

ARO WORKSHOP

**ANALYTICAL AND COMPUTATIONAL
ISSUES IN LOGISTICS R&D**

MAY 7-9, 1984



Volume 2

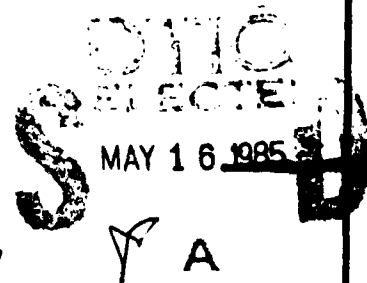
Approved for public release; distribution unlimited.
The findings of this report are not to be construed
as an official Department of the Army position, un-
less so designated by other authorized documents.

Sponsored by

The Army Research Office

Hosted By

The George Washington University



DTIC FILE COPY

AD-A153 853

28 4 10 079

U. S. Army Research Office

December 1984

PROCEEDINGS OF THE ARO WORKSHOP ON
ANALYTICAL AND COMPUTATIONAL
ISSUES IN LOGISTICS R&D

VOLUME 2

Sponsored by

The Mathematical Sciences Division

The Army Research Office

Host

Department of Operations Research

The George Washington University

Held at

Building C, Room 108

7-9 May 1984

CLASSIFIED	
DTIC TAB	
Unannounced	
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

Approved for public release; distribution unlimited.
The findings in this report are not to be construed
as an official Department of the Army position,
unless so designated by other authorized documents.

U. S. Army Research Office
P. O. Box 12211
Research Triangle Park, North Carolina

TABLE OF CONTENTS

VOLUME 2

<u>TITLE</u>	<u>PAGE</u>
Logistics, Stochastic Networks, and Failure Modeling Austin J. Lemoine and Michael L. Wenocur	287
Optimal Control of Admission, Routing, and Service in Queues and Networks of Queues: A Tutorial Review Shaler Stidham, Jr.	330
Some Recent Advances in Activity Networks S. E. Elmaghraby, B. M. Dodin, A. M. Salem, and A. H. Colby ...	378
Markov Models of Multi-echelon, Repairable-Item Inventory Systems Donald Gross, Leonidas C. Kioussis, Douglas R. Miller, and Richard M. Soland	421
Studying Model Assumptions in Process Quality Control C. P. Quesenberry	440
Recent Research in Experimental Design for Quality Improvement with Applications to Logistics George E. P. Box	454
Models of Military Combat with Logistics Components John S. Maybee	469
Overview of Expert Systems Stephen E. Cross	478

to the Department of Logistics, Army Research



LOGISTICS, STOCHASTIC NETWORKS, AND FAILURE MODELING

**AUSTIN J. LEMOINE
MICHAEL L. WENOCUR**

**Ford Aerospace & Communications Corporation
Palo Alto, California**

Logistics Systems designed to support

- **Availability**
- **Reliability**
- **Mission effectiveness**

Of Operating Systems in the field via

- **Supply**
- **Maintenance and Repair**

- **System**

- Operating ensemble of interest
(squadron of military aircraft)

- **Entity**

- Basic operating unit in system
(single aircraft)

- **Module**

- Basic repairable/replaceable unit for entity
(engine/components, navigational equipment,
communications equipment)

LOGISTICS AND AVAILABILITY

- Making provisions to have equipment ready when needed
- Good availability (high probability of being ready to operate) achieved by returning failed equipment to an operational state quickly

IMPROVING AVAILABILITY FOR A GIVEN DEFINED OPERATING SYSTEM

- Increase supplies of spare parts
 - Increase/Improve diagnostic and repair capabilities
- (Not entirely independent; downtime depends on spare parts needed for repair)

LOGISTICS AND RELIABILITY

- Satisfactory functioning of equipment when operated
- Depends upon
 - System design
 - Quality control in manufacturing
 - Checkout, maintenance, repair



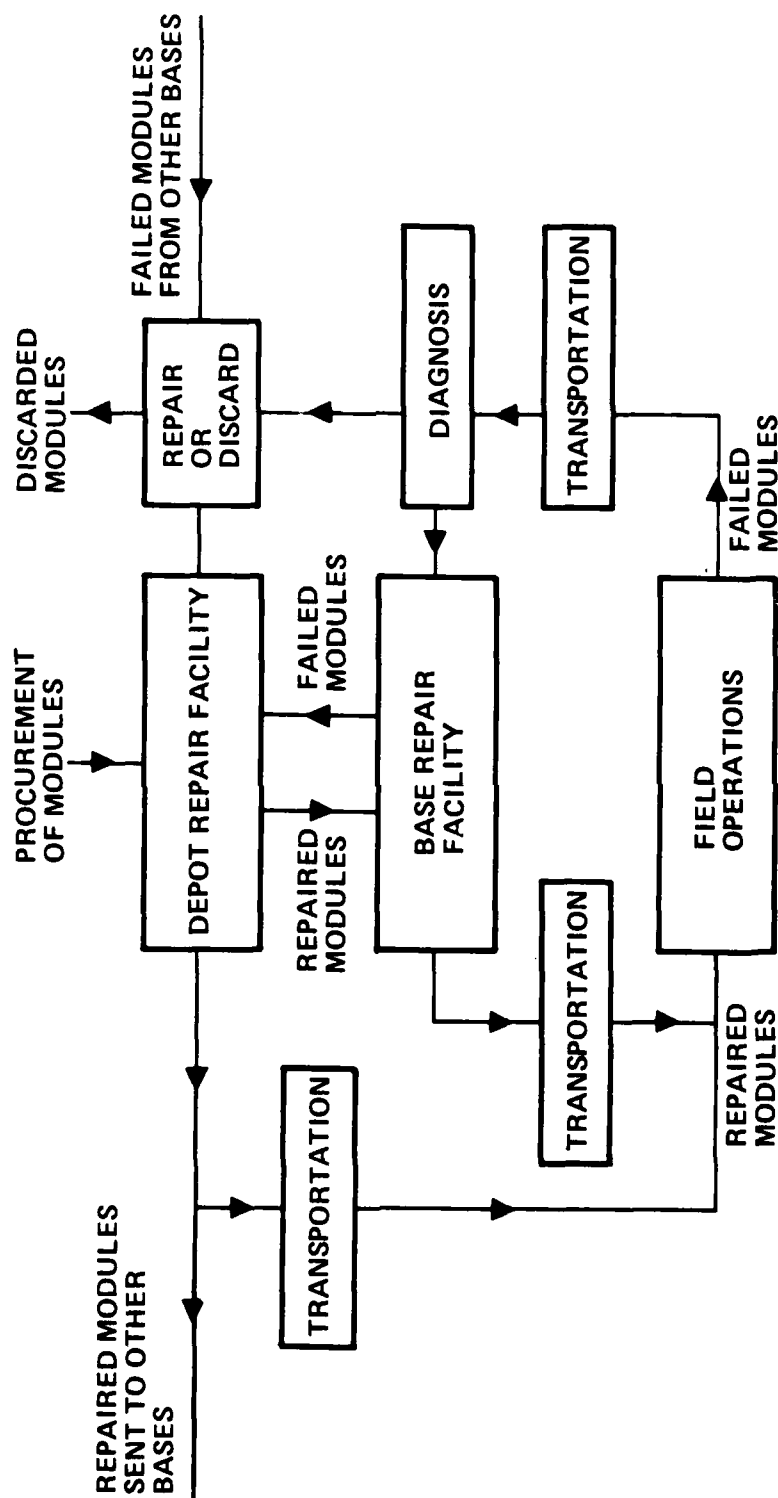
LOGISTICS AND MISSION EFFECTIVENESS

- Available equipment
 - Reliable equipment
 - Supply support
- } CRITICAL!

LOGISTICS SYSTEM

- Basic operating entities
- Spare-parts inventories
- Repair equipment
On-site (base), central depot
- Maintenance personnel
- Operating procedures
- Operating objectives
- System constraints

MODEL OF A LOGISTICS SYSTEM



MAIN PROBLEM

Maximize performance of the operating system
at the least logistic cost



SOME MEASURES OF PERFORMANCE

- System availability
 - (Steady-state probability system operable)
squadron of 50 aircraft
system operable when at least 40 ready to go
- System persistence
 - (Mean time between system failures)
system fails when fewer than 25 ready to fly



ATTACKING THE SYSTEM OPTIMIZATION PROBLEM

- “Bottom-Up” Approach
- “Top-Down” Approach



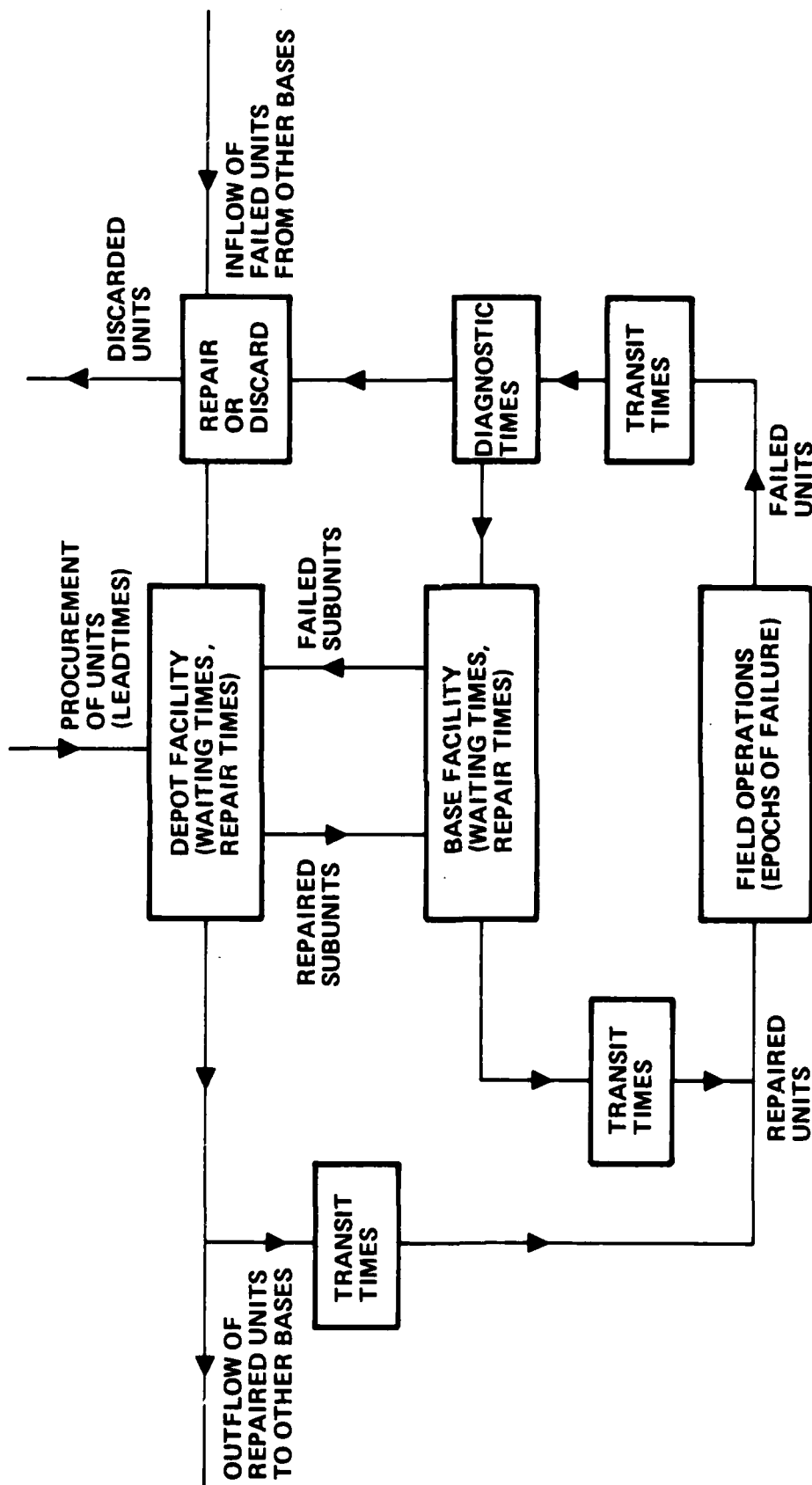
“BOTTOM-UP” APPROACH

- Build analysis from ground-level up
- “Add up” performance characteristics of individual logistics system components to predict overall system performance

“TOP-DOWN” APPROACH

- Start from a “system” point of view
- Model the more important interactions to begin with
- Refine the system model iteratively, both laterally and downward

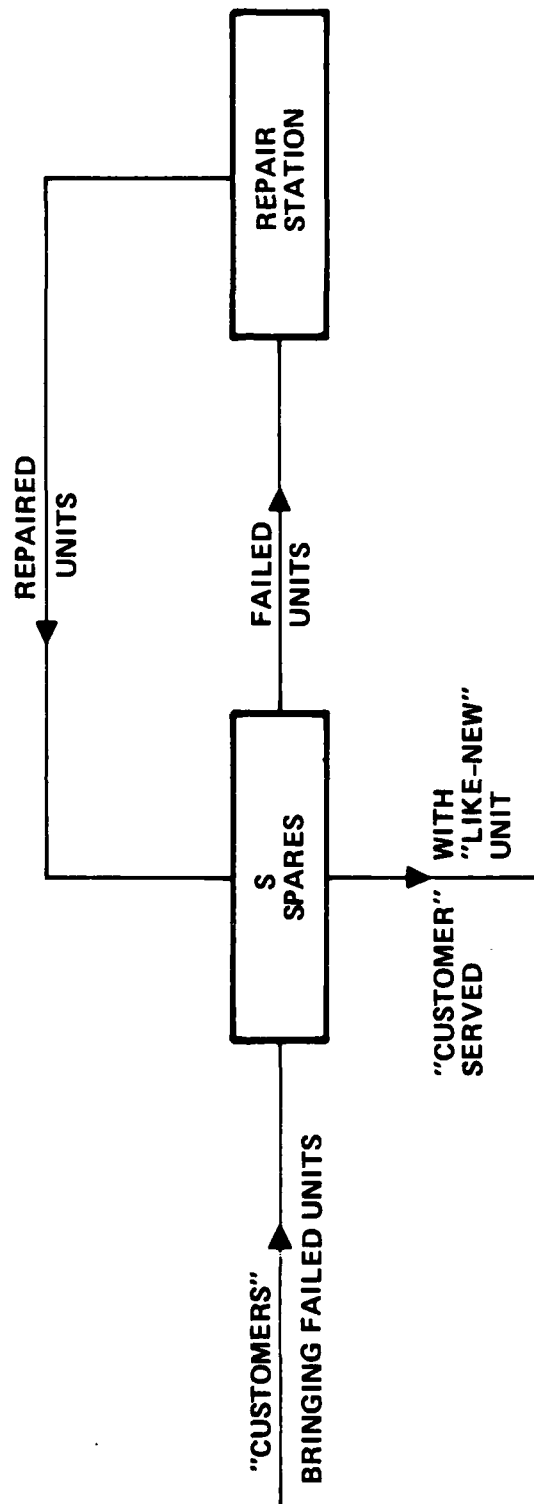
LOGISTICS SYSTEM/STOCHASTIC (QUEUEING) NETWORK



SOME TYPICAL QUEUEING MODELS ARISING IN LOGISTICS



INFINITE SOURCE MODEL

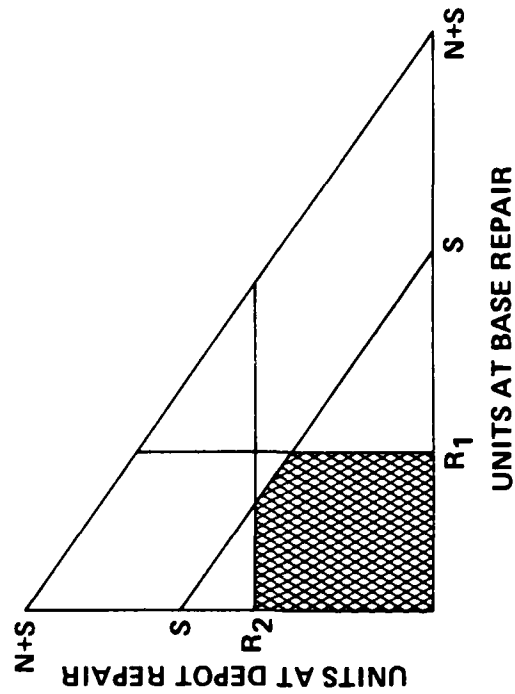
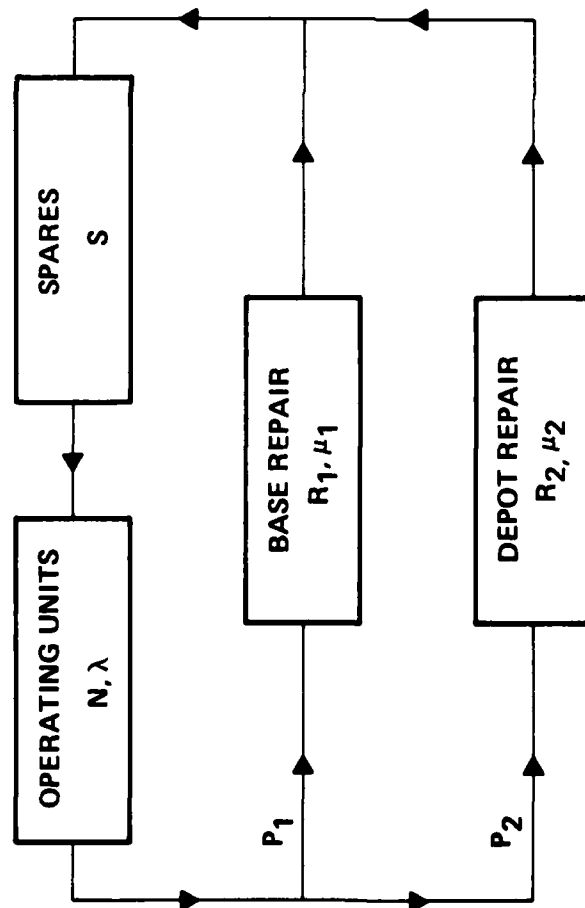


- INPUT STREAM
 - POISSON
 - RENEWAL

- REPAIR MECHANISM
 - REPAIR DISTRIBUTION
 - NUMBER OF SERVERS

REF: LUREAU (1974)

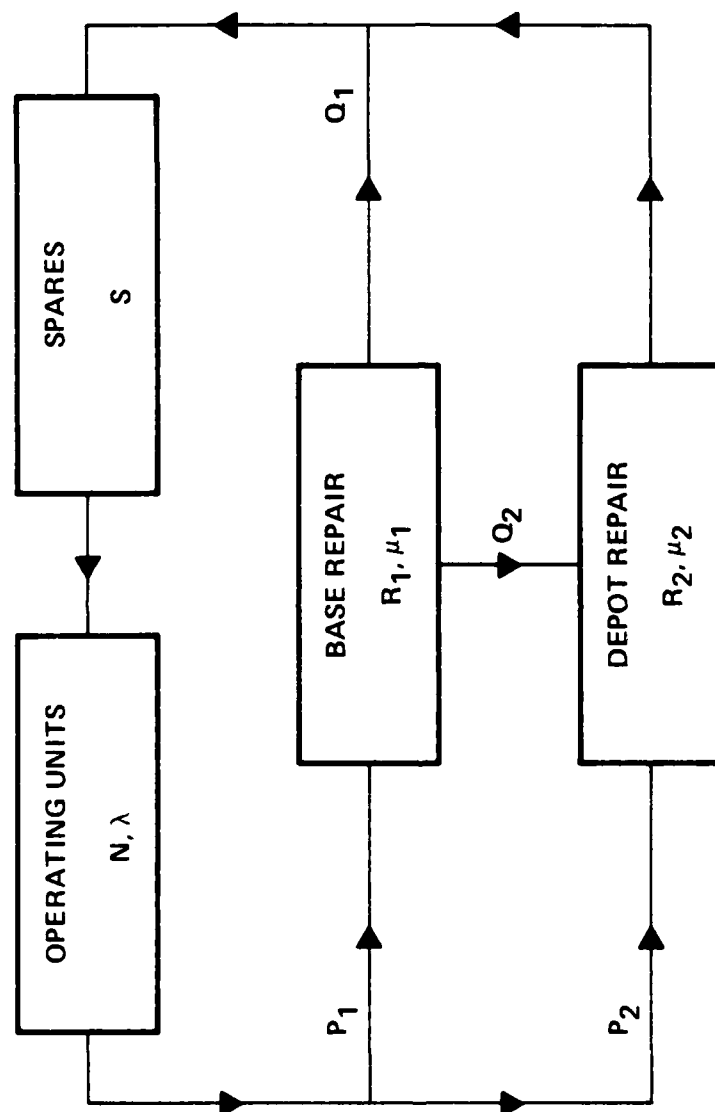
FINITE INPUT / ONE ITEM, TWO REPAIR FACILITIES



ALL FAILURE AND REPAIR TIMES EXPONENTIALLY DISTRIBUTED

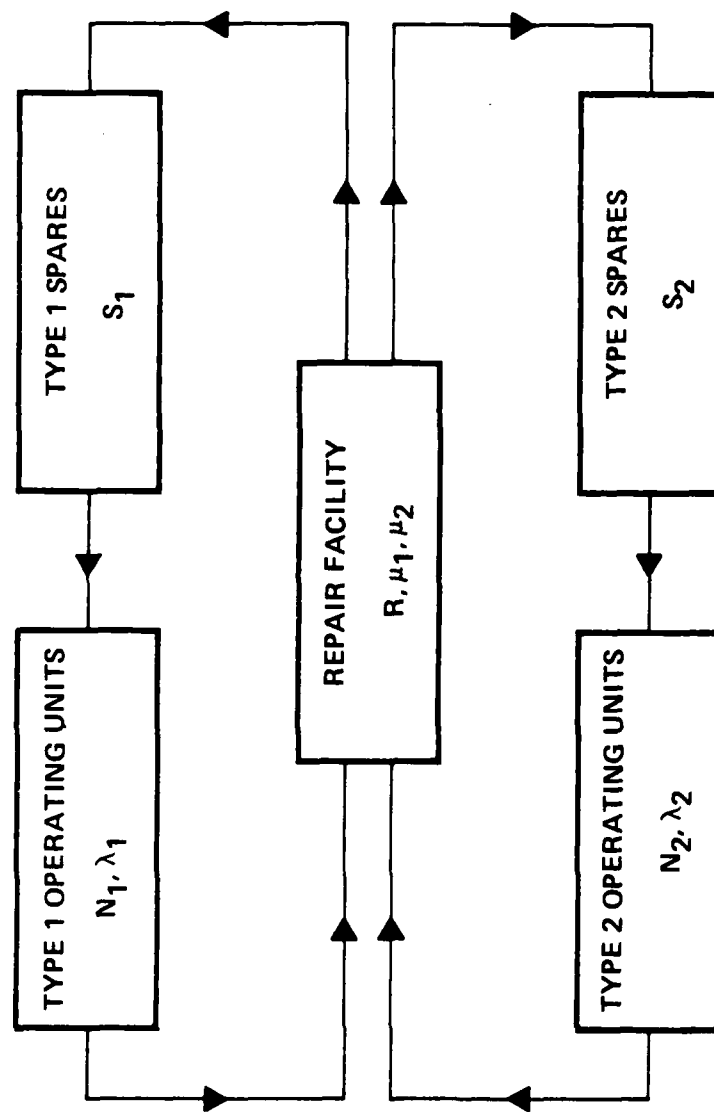
REF: IGLEHART AND LEMOINE (1973, 1974)

VARIATION OF FINITE INPUT / ONE ITEM MODEL



REF: IGLEHART AND LALCHANDANI (1973)

FINITE INPUT / TWO ITEM, ONE REPAIR FACILITY



ALL FAILURE AND REPAIR TIMES EXPONENTIALLY DISTRIBUTED
TYPE-1 UNITS HAVE PRE-EMPTIVE PRIORITY OVER TYPE-2 UNITS

REF: IGLEHART AND LALCHANDANI (1973)

ANALYSIS OF FINITE INPUT MODELS

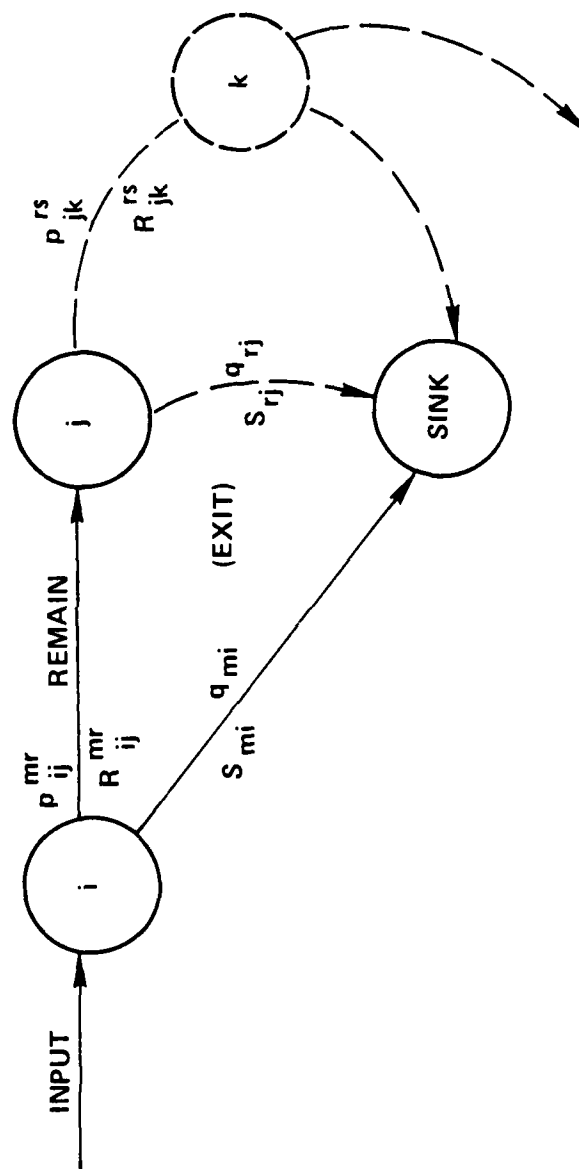
- View logistics system as closed network of queues
customers - units or modules
nodes - field sites and repair facilities
- Vector process tracking number of units at each
node, type, and position in queue, MARKOV
- Calculate stationary (steady-state) distribution
ref: Lemoine (1977), Kelly (1979)
- Assess performance
ref: Lureau (1974), Lemoine (1978)

STOCHASTIC NETWORK MODEL

Ref: Lemoine (1983)

NETWORK MODEL

N NODES $i, j, k \in \{1, 2, \dots, N\}$
 M OBJECT CLASSES $m, r, s \in \{1, 2, \dots, M\}$



PLAUSIBLE MODEL FOR COMPLEX SEQUENTIAL PROCESS WHEREIN OBJECT OR "SYSTEM"

- Moves through succession of states (nodes)
and operating modes (classes)
- Transitions among states and operating modes
 - Occuring in possibly random manner
 - Requiring resources in randomly varying amounts



SYSTEM LIFE CYCLE

Availability  Unavailability

- Readiness
 - Fully operational
 - Degraded
- Failure
 - Random
 - Design defects, QC
- Repair/maintenance
 - Restorative
 - Inadequate/damaging
 - Skill levels
 - Support equipment
 - Spares provisioning
 - Faulty diagnostic and built in-test equipment

MODEL RESULTS

- Routing
First-passage and exit probabilities
- Resources
Requirements distributions: global and nodal
- Poisson external input/"infinite-server" nodes
Time-dependent (Π_t) and
Stationary (Π) distributions of state

$d(\Pi_t, \Pi)$

Confidence bounds on number of occupied servers

MODELING STATUS OF LOGISTICS/REPAIR SYSTEM

Many units of diverse types

Variety of support functions
(Maintenance, Repair, Replenishment, etc)

Pooled process of initial failures Poisson

Stochastic network model

- Approximate status of overall system,
account for numerous important factors
- Assess adequacy of support resources
vis a vis mission requirements and
operational availability

LOGISTICS AND FAILURE MODELING

Reliability modeling for complex systems important for:

- Predicting and analyzing failures
- Structuring operating policies
- Determining logistics support requirements
- Influencing design decisions
- System acquisition strategy

Improved failure modeling can have significant impact on structuring life-cycle decisions

FAILURE MODELING

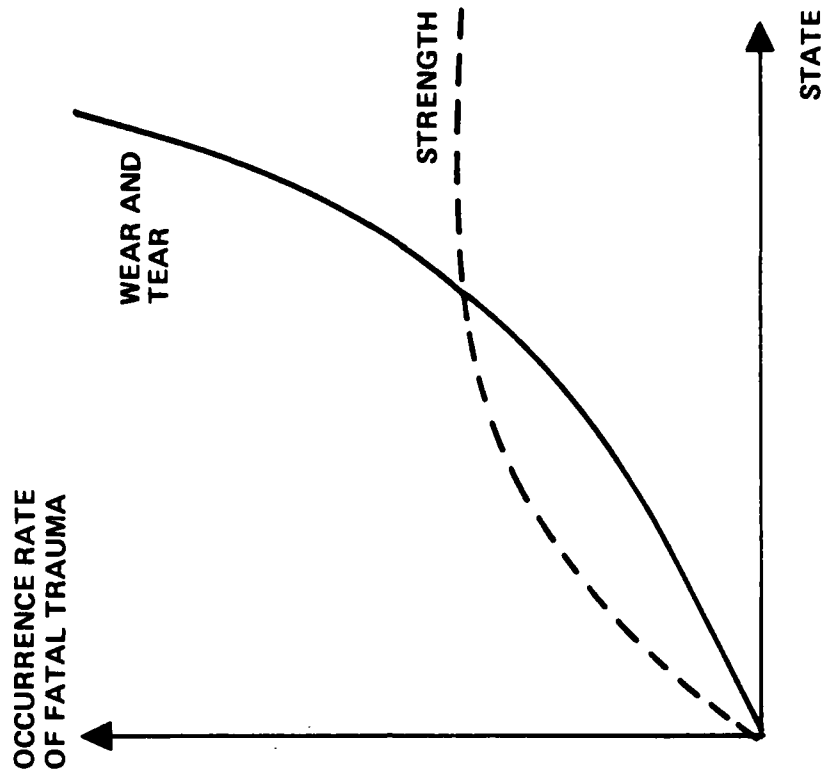
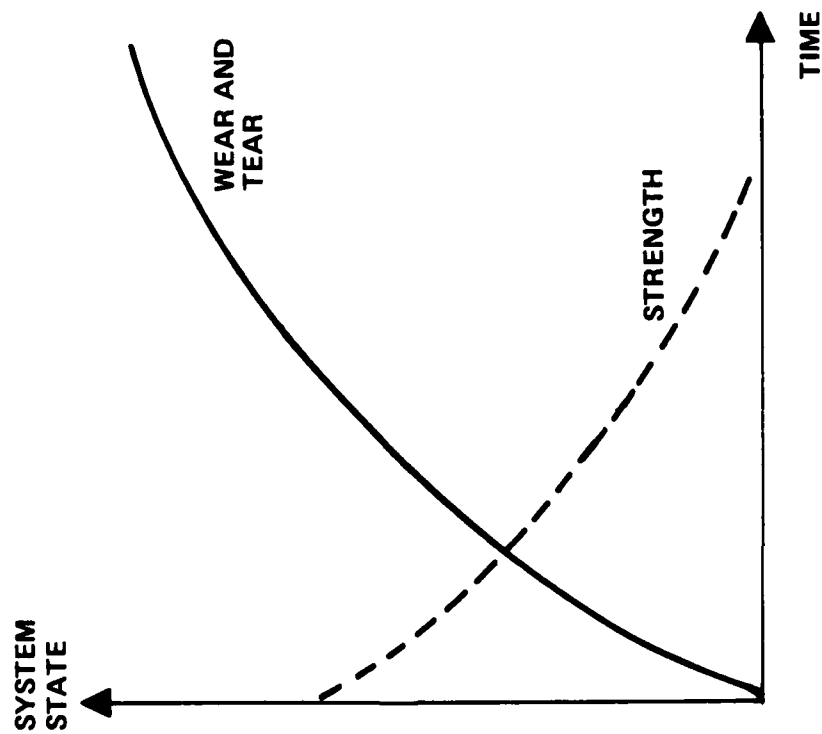
Take a Process Point of View

- Capture the dynamic dependency between system state and system reliability
- Reflect the dynamic dependency of system failure and decay on the state of the system



**Ford Aerospace &
Communications Corporation**

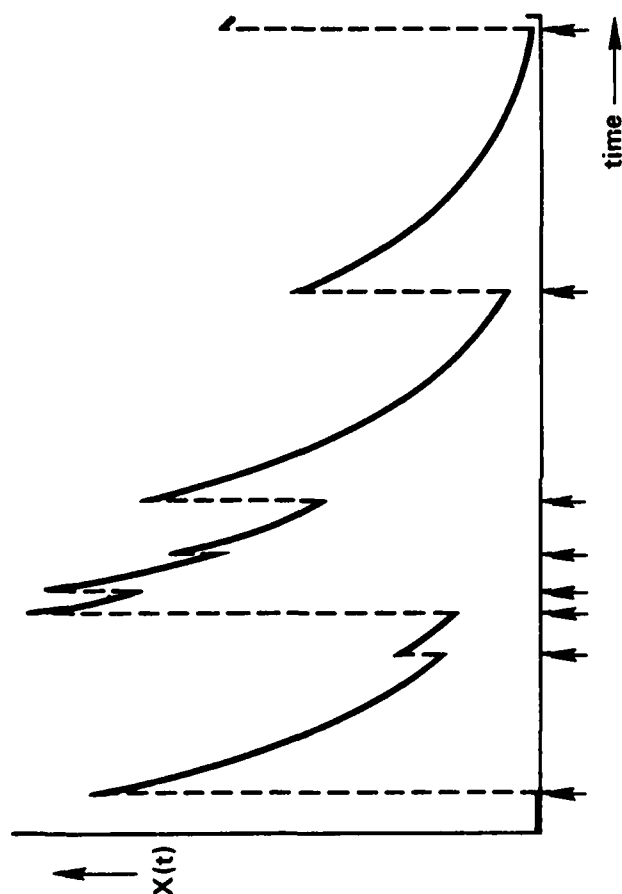
SYSTEM STATE AND SUSCEPTIBILITY TO FAILURE



TWO RELEVANT APPROACHES

- Shot noise model for system stress
- Inverse Gaussian distribution

SHOT NOISE MODEL



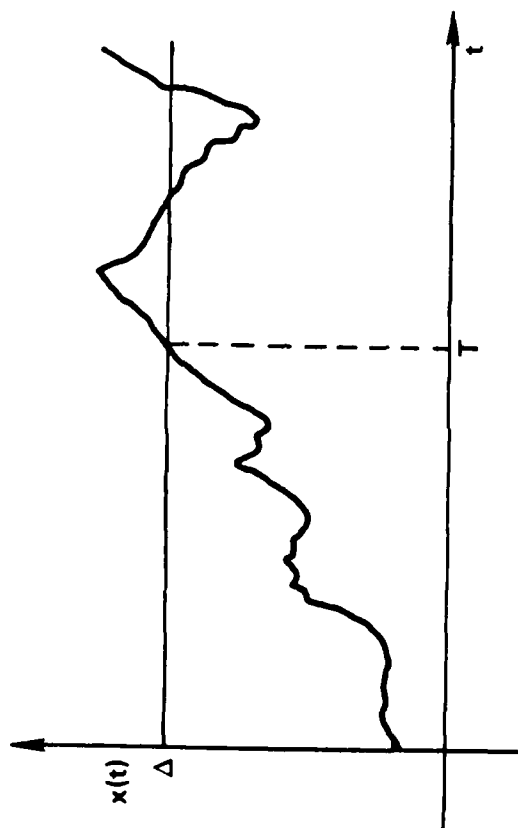
$$X(t) = \sum_{S_n \leq t} D_n e^{-(t - S_n)}$$

POISSON TRAUMA EPOCH

SEVERITY OF TRAUMA

AFTEREFFECT/
RATE OF RECOVERY

INVERSE GAUSSIAN DISTRIBUTION



Δ CRITICAL (FAILURE) THRESHOLD FOR WEAR AND TEAR

$x(\cdot)$ BROWNIAN MOTION WITH DRIFT

T FAILURE TIME = EPOCH OF FIRST PASSAGE TO Δ
INVERSE GAUSSIAN DISTRIBUTION

AN APPROACH TO FAILURE MODELING

$X(\cdot)$ STATE PROCESS (WEAR, STRENGTH)

$k(\cdot)$ KILLING RATE FUNCTION

Δ CRITICAL (FAILURE) THRESHOLD

τ EPOCH OF FIRST PASSAGE TO Δ

T EPOCH OF SYSTEM FAILURE

$$P^X \{ T > t \} = E^X \left\{ \exp \left[- \int_0^t k(X(s)) ds \right] \cdot 1_{\{ \tau > t \}} \right\}$$

DIFFUSION MODEL FOR X(•)

$$dX(t) = \mu(X(t))dt + \sigma(X(t))dB(t)$$

$$w(x, t) = P^x \{T > t\}$$

$$\frac{\partial w(x, t)}{\partial t} = -k(x)w(x, t) + \mu(x) \frac{\partial w(x, t)}{\partial x} + \frac{\sigma^2(x)}{2} \frac{\partial^2 w(x, t)}{\partial x^2}$$

CRITICAL LEVEL $\Delta, 0 \leq \Delta \leq +\infty$

X(•) STRENGTH, $w(x, 0) = 1$ if $x > \Delta$

X(•) WEAR AND TEAR, $w(x, 0) = 1$ if $x < \Delta$

MODEL SUBCLASSES AND PARAMETERS

SUBCLASS	INFINITESIMAL PARAMETERS $\mu(x), \sigma^2(x)$	KILLING FUNCTION $k(x)$	FAILURE THRESHOLD Δ
DETERMINISTIC WEAR	$\sigma^2(x) \equiv 0$	ARBITRARY	ARBITRARY
CONSTANT KILLING RATE	ARBITRARY	$k(x) \equiv \lambda$	$\Delta < +\infty$
INFINITE LEVEL OF WEAR AND TEAR	ARBITRARY	ARBITRARY	$\Delta = +\infty$

DETERMINISTIC SYSTEM STATE PROCESS

$$\sigma(x) \equiv 0$$

$$X(x) = \int_0^t \mu(X(s))ds + x$$

$$P^x \{T > t\} = \exp \left\{ - \int_0^t k(X(s))ds \right\} 1_{\{t > 0\}}$$

$$\Delta = +\infty, \quad x = 0$$

$$P \{T > t\} = \exp \left\{ - \int_0^t r(s)ds \right\}$$

FAILURE RATE FUNCTION

$$r(t) = k(X(t))$$

IMPLICATIONS

- MAKEHAM'S LIFE DISTRIBUTION
GOMPERTZ LIFE DISTRIBUTION

$$k(x) = x \quad \mu(x) = c(x - b) \\ b = 0$$

- WEIBULL DISTRIBUTION
EXPONENTIAL DISTRIBUTION

$$k(x) = x^{b-1}(b+1) \quad \mu(x) = c \frac{1}{b-1} \\ k(x) = \lambda \quad \text{or} \quad \mu(x) = 0$$

- RAYLEIGH DISTRIBUTION

$$k(x) = x \quad \mu(x) = b$$

- GUMBEL DISTRIBUTION

$$k(x) = \frac{x}{b(e^x - 1)} \quad \mu(x) = -\frac{x}{b}$$

FIRST-ORDER MODEL ($\sigma(x) = 0$) YIELDS CLASSICAL DISTRIBUTIONS!



Ford Aerospace &
Communications Corporation

NON-DETERMINISTIC WEAR (STRENGTH) MODELS

– $X(\cdot)$ DIFFUSION PROCESS

– $X(\cdot)$ SHOT-NOISE PROCESS

DIFFUSION MODELS FOR WEAR

- FIRST PASSAGE TO Δ WITH CONSTANT KILLING RATE

$$p^x \{T > t\} = e^{-\lambda t} p^x \{ \tau > t \}$$

- FUNCTIONS OF BROWNIAN MOTION

- GENERALIZED INVERSE GAUSSIAN

- FAILURE DUE TO SHOCK ONLY

- BROWNIAN MOTION WITH QUADRATIC KILLING



Ford Aerospace &
Communications Corporation

DIFFUSION MODELS FOR WEAR (CONTINUED)

$$X(t) = \sigma B(t) + \mu t + x$$

$$k(y) = \lambda y^2$$

$$P^x \{T > t\} = \left[\text{SECH} \left(\sigma t \sqrt{2\lambda} \right) \right]^{1/2} \text{EXP} \left\{ -\frac{\mu^2 t}{2\sigma^2} - \frac{x\mu}{\sigma^2} \right\} \cdot \psi(x, t)$$

$$\psi(x, t) = \text{EXP} \left\{ \frac{\mu^2}{2\sigma^3 \sqrt{2\lambda}} \text{TANH} \left(\sigma t \sqrt{2\lambda} \right) + \frac{x\mu}{\sigma^2} \text{SECH} \left(\sigma t \sqrt{2\lambda} \right) - \frac{x^2}{\sigma} \sqrt{\frac{\lambda}{2}} \text{TANH} \left(\sigma t \sqrt{2\lambda} \right) \right\}$$

IFR DISTRIBUTION

SHOT-NOISE MODEL FOR STRESS

$$X(t) = \sum_{n=1}^{\infty} D_n h(t - S_n)$$

EPOCH OF n TH JOLT
 "RECOVERY" FUNCTION
 MAGNITUDE OF n TH JOLT

$$P\{T > t\} = \text{EXP}(-\lambda t) \text{EXP} \left\{ \lambda \int_0^t \left(\int_0^y h(z) dz \right) dy \right\}$$

IFR DISTRIBUTION

SUMMARY

- Stochastic networks
- Failure modeling

Have a Definite Role in Logistics!



OPTIMAL CONTROL OF ADMISSION, ROUTING, AND SERVICE
IN QUEUES AND NETWORKS OF QUEUES: A TUTORIAL REVIEW

Shaler Stidham, Jr.
North Carolina State University at Raleigh

September, 1984

Abstract

Queueing models can be useful in the analysis, design, and control of production, transportation, communication, and logistics systems. Using the theory of Markov decision processes and the inductive techniques of dynamic programming, normative models have been developed for optimal control of admission, routing, and servicing of jobs in queues and networks of queues. We review some of these models in a unified format, beginning with single-facility models and then moving on to models for networks of queues. The emphasis is on using induction (value iteration) to establish the qualitative structure of optimal control policies. We compare the resulting policies to some ad hoc control rules that have been proposed in the literature.

Address: Department of Industrial Engineering and
Program in Operations Research
Box 7906
North Carolina State University
Raleigh, NC 27695-7906

Acknowledgement:

This research was partially supported by the U.S. Army Research Office under Contract DAAG29-82-K-0152 at N.C. State University.

several stages of disassembly, inspection and/or replacement of components. In communication systems, a series of queues may provide a model for a "virtual channel" -- a sequence of linked communication channels -- between the source and destination of a certain class of messages (cf. Lazar(1983)).

It may be the case that different jobs require different combinations of services in different sequences. There could be feedback of certain jobs, because of the need for rework. In such cases the appropriate queueing-network model will be more complicated, with multiple branches and combined series-parallel structure. The classical industrial application for such a more general network model is a job shop, and indeed the seminal theoretical paper on networks of queues (Jackson(1963)) has these words in the title. Subsequently, Jackson's model and generalizations have been successfully applied to performance evaluation in computer/communication systems (see, e.g., Kleinrock(1976)). More recently, the industrial engineering community has recognized the utility of the networks-of-queues model for analysis of flexible manufacturing systems: systems consisting of automated manufacturing cells, capable of efficiently processing a variety of jobs requiring processing by different combinations of machines (see, e.g., Buzacott and Yao(1983)).

For the most part, queueing-network models have been descriptive, rather than normative. That is, they have provided a tool for estimating operating characteristics or measures of effectiveness, such as congestion levels or throughput, of

existing or proposed systems, operating according to specified policies. The task of selecting the best design or the best control policy has been left to the system operator. Recently, however, more and more researchers have turned to mathematical models, not only for description, but also to help make crucial design and control decisions.

Optimal design is an integral part of queueing-network models for automatic transfer lines, where the location and size of buffers for in-process inventories are design variables (Ho et al(1979), Altioek(1982), Altioek and Stidham(1983)). A common approach is to use simulation or an analytical Markov model to evaluate the costs and benefits of a fixed buffer configuration, and then use a gradient-search algorithm to move toward a local (and, with luck, global) optimum. Other possible design variables are the number of servers and/or the service rates at each node (e.g., work station, repair facility) of the network. Gross, Miller, and Soland(1982) used a queueing-network model to represent a system in which repairable items are processed at either a depot or field station and there is the possibility of providing spares. The number of spares and the number of servers at each repair facility were design variables and the objective was to minimize the cost of providing spares and servers, while maintaining a desired level of operational readiness (probability that the required number of items are operational). An integer-programming algorithm was used to find the optimal design.

In a design problem, the decision variables, once fixed,

remain so throughout the planning horizon. By contrast, if the decision variables can be adjusted as the system status changes, then we have a control problem. Control may be exercised in a queueing network by varying the arrival or service rates, turning servers on or off, or changing job priorities or routings. By doing so, one can balance the "bad" (congestion) with the "good" (throughput). As an example, consider the communication system illustrated in Figure 2. (This example captures, in simplified form, several of the important issues in the control of flow and routing in communication systems.)

Figure 2. Simple Communication Example.

There are three cities (A, B, and C), with direct channels labelled 1, 2, and 3, linking A to B, B to C, and A to C, respectively. Each channel transmits messages one at a time. Messages waiting to be transmitted are placed in an infinite-capacity buffer in front of the channel. There are three classes of messages (jobs): originating in A and destined for B (class 1), originating in B and destined for C (class 2), and originating in A and destined for C (class 3). The system has no control over messages of class 1 or 2: all messages of each class must be sent over the corresponding direct channel. Messages of class 3, however, may be controlled in two ways: (i) by accepting or rejecting a message when it "arrives" (is generated at city A); (ii) by choosing whether to route it directly to city C via channel 3, or indirectly via channels 1

and 2. In the latter case, the message competes with "local" traffic (messages of classes 1 and 2).

The corresponding network of queues (see Figure 3) has three

Figure 3. Queueing-Network Model for Communication System.

nodes, one corresponding to each channel (server) and its buffer. The decisions (accept vs. reject, route via channels 1 and 2 vs. via channel 3) are indicated by "toggles".

What can one say about "optimal", or at least "good", control policies for this problem? On the one hand, one would like to get as many messages as possible through the system (maximize throughput). This could be accomplished by admitting each class-3 arrival. On the other hand, admitting a class-3 message has associated "costs". One cost is reflected in the time it takes the message to reach its destination, which depends on which route is chosen and how many messages are ahead of it in the buffer(s). Economists would call this an "internal effect". Another cost is the "congestion" which is added to the system by admitting the message, as reflected in the additional delays suffered by later messages because of its presence. The economic term for this type of cost is an "external effect". A rational control policy for admission and/or routing of messages must balance these benefits and costs. Intuition suggests that such a policy should have (at least) the following (monotonicity) properties: if it accepts an arriving message in a given state, then it should also accept if one or more messages are removed

Often it is useful to view the components of production, transportation, communication, or logistics systems as queues, in which jobs (customers) are processed by one or more servers. The jobs could be parts or subassemblies, vehicles, messages or computer programs, or repairable items. The servers could be work stations, traffic signals or road segments, communication channels or computer CPU's, or maintenance/repair facilities. Often there are many such service facilities, linked together by paths along which jobs may travel from one facility to another.

An abstract model for such systems is a network of queues, and such models have been increasingly recognized as useful tools for understanding the behavior of complex service systems. Perhaps the simplest network consists of a number, m , of queues in series, in which the output of queue i is the input to queue $i+1$. A flow-shop production system, such as an assembly line or automatic transfer line, has this structure (see Figure 1). In

Figure 1. Queues in Series.

such a system, each job must be processed at each of the work stations (numbered $i = 1, 2, \dots, m$) in the same order. Each work station consists of one or more servers (machines and/or workers) all capable of performing the same task, preceded by a buffer or storage area where jobs wait for processing. Obviously, such a model is not limited to production and assembly operations, but may also apply to maintenance and repair facilities in which incoming jobs (e.g., aircraft, vehicles) of identical or similar configuration scheduled for routine maintenance must go through

from any one of the nodes; if it is preferable to route a class-3 message via node 3 in a given state, then it will remain preferable to do so if one or more messages are removed from node 3 and/or added to nodes 1 and 2.

At the very least, one expects a mathematic control model to be capable of confirming or denying the validity of such intuition. Beyond that, such a model should lead to efficient numerical algorithms for computing the control parameters of an optimal policy -- specifically, the boundary of the "acceptance region" and the boundary between the regions of the state space where it is optimal to route to nodes 1 and 2 vs. node 3 (the "switching curve").

For another example of a control problem involving a network of queues, let us return to the series-of-queues model for a flow shop as pictured in Figure 1. Suppose that it is possible to control (dynamically vary) the rates at which the servers at various nodes work, in response to changing congestion levels. For example, one might want to turn a server off when the downstream buffer(s) have accumulated a large number of jobs or when the number of jobs in the upstream buffer(s) is small. "Just-in-time" production policies like the Japanese *kanban* are a special case of this type of policy. Again, one would hope that a mathematical optimization model, based on a plausible benefit-cost structure, would enable one to test the validity of such operating principles and efficiently calculate the parameters of the associated control policies.

The primary goal of this paper is to survey the progress that has been made toward accomplishing these two goals, in the general context of mathematical models for control of networks of queues. We focus attention on models based on Markov decision processes, using the inductive ideas embodied in dynamic programming as a tool for characterizing the structure of optimal policies as well as calculating their parameters. We begin by illustrating the economic assumptions and analytic technique in the context of some simple, one-facility models, and then move on to multi-facility (queueing-network) models. The models considered allow control of admission, routing, and/or servicing of jobs. For a more detailed survey that concentrates on control of admission and routing, see Stidham(1984). Earlier comprehensive surveys may be found in Sobel(1974), Stidham(1974), and Crabill, Gross, and Magazine(1977).

1. Admission Control to a Single-Server Queue

We first consider a simple, single-facility model with control of admission of customers. It is a special case of a model for exponential congestion systems, studied by Lippman and Stidham(1977) and is illustrated below in Figure 4. Jobs arrive

Figure 4. M/M/1 Queue with Control of Admission.

at a single-server facility according to a Poisson process with mean arrival rate λ (jobs per unit time). Equivalently: the interarrival times between jobs are i.i.d. (independent and iden-

tically distributed) with an exponential distribution with mean $1/\lambda$.) (See, e.g., Ross(1970) for a discussion of properties of the Poisson process and exponential distribution.) The system operator controls the arrivals by deciding whether to accept (action $a = 1$) or reject (action $a = 0$) an arriving job. Accepted jobs join an infinite-capacity queue and wait for service. There is a single server who serves jobs one at a time, with service times that are i.i.d. with an exponential distribution with mean $1/\mu$. The shorthand for this is to say that we have an exponential server with mean service rate μ (jobs per unit time). In the literature on queues, a system like that illustrated in Figure 4, but with no restriction on entry of jobs, is referred to as an M/M/1 queue. The "M" in the first position stands for the "memoryless" (exponential) distribution of interarrival times. The "M" in the second position tells us that the service-time distribution is also exponential. The "1" in the third position stands for "one server". We extend this terminology to control models, so that the model under consideration becomes an "M/M/1 queue with control of admission".

For clarity of exposition, we assume a simple benefit/cost structure reflecting the fact that throughput is "good" and congestion is "bad". Each admitted job generates a fixed reward (utility) r . There is a waiting cost h per job per unit time in the system (i.e., in the queue plus in service). Equivalently, by analogy with inventory-control problems, we can say that there is a cost of holding jobs $h \cdot i$ per unit time while there are i jobs in the system. Future rewards and costs are

continuously discounted at rate $\alpha > 0$, so that the present value of a net benefit x received at time t is $x \cdot \exp(-\alpha t)$. The objective of the system operator is to maximize the total expected discounted net benefit over an infinite time horizon.*

 *Discounting reflects the time preferences of a rational economic decision maker and makes it possible to compare present and future benefits and costs. An alternate optimality criterion is long-run average net benefit per unit time. Average-optimal control policies can be derived from α -discount-optimal policies by letting α approach zero (see, e.g., Ross(1970), Lippman and Stidham(1977)). We shall therefore confine our attention to the discounted-net-benefit criterion.

Our goal is to characterize the structure of an optimal control policy (rule for choosing actions) and develop efficient techniques for computing its parameters. To this end, we first use concepts from dynamic programming (Bellman(1958), Ross(1970)) to derive a functional equation satisfied by the optimal value function, $V(i) :=$ maximum total expected discounted net benefit over an infinite horizon, starting from state i . The Principle of Optimality of dynamic programming says that, from each starting state i , an optimal policy will choose an action $a(i) = a$ that maximizes the sum of the discounted net benefit earned until the next observation point and the present value of the discounted net benefit earned after the next decision point, assuming that we follow an optimal policy from whatever state j we enter at that point. For the problem at hand, it follows that the optimal value function $V(i)$ satisfies the optimality equation (1.20) :

$$V(i) = [-h \cdot i + \lambda \max\{r + V(i+1), V(i)\} + \mu V(i-1)] / (\lambda + \mu + \alpha), \quad (1)$$

where $V(-1) = V(0)$. A rigorous derivation of the dynamic-

programming optimality equation under general conditions satisfied by our problem may be found in Schal(1975), Bertsekas(1980), or Whittle(1983). For the problem at hand, there are several possible heuristic derivations of (1), one of them being the following.

We observe the system only at transitions (changes of state caused by arrivals or service completions). The time between transitions is the minimum of two independent exponential random variables, the time until the next arrival and the time until the next service completion, and thus is itself exponentially distributed with parameter $\lambda + \mu$, the sum of the arrival and service rates. Until a transition occurs, given that we are in state i , we incur holding cost at the constant rate $h \cdot i$. Applying the discount factor $\exp(-\alpha t)$ for cost accruing at time t and integrating over the exponentially distributed time until the next transition gives the term $-h \cdot i / (\lambda + \mu + \alpha)$, the expected discounted holding cost until the next observation point. The expected discount factor over the interval between now and the next observation point is $(\lambda + \mu) / (\lambda + \mu + \alpha)$. The next observation is at an arrival or a service completion with respective probabilities $\lambda / (\lambda + \mu)$ and $\mu / (\lambda + \mu)$. These expressions follow from well-known properties of exponential distributions.*

*Note that because $V(-1) = V(0)$, the optimality equation for state 0 implicitly assumes that the server continues to operate when no customers are present, but that any service completions are fictitious, in effect creating "dummy" transitions from state 0 to itself. It is easy to verify that the equation for state 0 is equivalent to

$$V(0) = \lambda \max(r + V(1), V(0)) / (\lambda + \alpha),$$

the equation resulting from imbedding only at "real" events (arrivals). The advantage of our formulation (often referred to in the literature as "uniformization") is that it makes the time between observation points (although random) independent of the state and action. The resulting optimality equations are structurally equivalent to those of the approximating discrete-time Markov decision process resulting from observing the system at fixed time intervals of length dt and ignoring multiple events in a single interval (which have probability $o(dt)$). The equations in this form are more amenable to qualitative analysis via successive approximations, as we shall see presently. Uniformization is a standard technique in the analysis of Markov chains. Its usefulness in models for control of queues was recognized by Lippman(1975) (see also Serfozo(1979)).

We can use the optimality equation to show that an optimal policy is *monotonic*, specifically that the optimal action $a(i)$ is non-increasing in $i \geq 0$. In other words, an optimal admission policy is characterized by a *critical number* i^* such that an arriving job is admitted if and only if $i < i^*$. By adding and subtracting the term $V(i)$ on the right-hand side, we can rewrite the optimality equation (1) in equivalent form

$$V(i) = [-h \cdot i + \lambda V(i) + \mu V(i-1) + \lambda \max(r - [V(i) - V(i+1)], 0)] / (\lambda + \mu + \alpha), \quad (2)$$

from which it is easy to see that an optimal admission policy will be monotonic if $V(i) - V(i+1)$ is non-decreasing in i , that is, if $V(i)$ is concave. In this case, the critical number will be $i^* = \min\{i: V(i) - V(i+1) \geq r\}$. The quantity $V(i) - V(i+1)$ can be interpreted as the total cost (including loss of net benefit to future arrivals) caused by the entry of a job in state i . We should admit a job if and only if this cost is smaller* than the reward r .

*Our convention is to reject in case cost exactly equals reward.

Since the optimality equation (2) only defines $V(i)$ implicitly, it does not yield a direct proof that $V(i)$ is concave. One can, however, exploit the fact that (2) can be solved by successive approximations: start with an approximation, V_0 , of the optimal value function V and insert it in place of V in the right-hand side of (1), thus generating a new approximating function V_1 . Repeat this process, defining V_n in terms of V_{n-1} ($n \geq 1$) by the recursive analogue of (1):

$$V_n(i) = [-h \cdot i + \lambda \max(r + V_{n-1}(i+1), V_{n-1}(i)) + \mu V_{n-1}(i-1)] / (\lambda + \mu + \alpha) . \quad (3)$$

The theory behind the convergence of V_n to V , including necessary restrictions on V_0 , may be found in Schäl(1975), Stidham(1981), Whittle(1983), van der Wal(1981). In dynamic programming, it is customary to call this approach *value iteration*, following Bellman(1957). V_n can also be interpreted as the maximum total expected net benefit if the system is to be operated for only n stages (observation points) and then shut down, earning a terminal (scrap) value according to a given function $V_0(j)$ of the final state j .

Value iteration can be used as the basis for an inductive proof that V_n , and hence V , is concave. It is intuitively clear that $V_n(i)$ should also be non-increasing in i . For technical reasons, having to do with the boundary at state $i=0$, it is necessary to add this property to the induction hypothesis, which thus becomes:

$$V_n(i) \text{ is concave and non-increasing in } i \geq 0. \quad (4)$$

The inductive step involves showing, via equation (3), that V_n satisfies (4) whenever V_{n-1} does. The non-trivial part is showing that concavity of a function $g(i)$ implies concavity of the function

$$f(i) := \max(r + g(i+1), g(i)). \quad (5)$$

This was done by Lippman(1975). The function $V_0 \equiv 0$ trivially satisfies (4) and therefore is a suitable starting function for the induction. Convergence of V_n to V in this case follows from Schäl(1975). In the process of verifying the optimality of a monotonic policy for the infinite-horizon discounted problem, we also verify monotonicity for each n -stage problem -- a property that may be of interest in applications to problems with a finite planning horizon.

Inductive analyses like this, based on the "preservation of concavity" through transformations like (5), form the basis for the study of the structure of optimal policies in many problems in the control of queues. The next section provides another illustration of the power of this approach.

Remarks.

1. Critical-number policies and their numerical computation.
The inductive analysis has reduced the M/M/1 admission-control problem to one of finding the optimal critical number i^* . This is a one-dimensional optimization problem: among the class of

M/M/1/n systems, find the system capacity $n = i^*$ that yields the maximum value. Naor(1969) used this approach in his seminal paper on admission control. For M/M/1/n systems a closed-form expression is available for the long-run average net benefit, which can be shown to be unimodal in n , so that a local maximum is a global maximum. Naor exploited this property to give necessary and sufficient conditions for $n = i^*$, when the optimality criterion is long-run average net benefit. This approach was extended to M/M/c and more general exponential systems by Knudsen(1972) and Knudsen and Stidham(1976). Systems with more general arrival process or service-time distribution, with attention restricted to critical-number policies, have been studied by Adler and Naor(1969), Simonovits(1976), Balachandran and Schaefer(1979), and Rue and Rosenshine(1981).

Of course, for discounted problems one possible technique for finding i^* is simply to apply value iteration as a numerical algorithm to the optimality equation (1).*

*The state space must first be truncated to a finite set, in order for the algorithm to be finite.

particularly fast algorithm by itself (especially when ρ is close to zero), value iteration can be made to converge quite rapidly with the incorporation of bounds, extrapolations, elimination of suboptimal actions, and transformations. (See, e.g., van Nunen and Wessels(1979) for a survey of the different variants of value iteration.) Policy iteration ("Howard's algorithm") is another alternative. A variant of policy iteration, which restricts attention to critical-number policies and ex-

exploits the special structure of the optimality equation for admission-control problems, was developed by van Nunen and Puterman(1980),(1982). Wijngaard and Stidham(1983) have developed an efficient algorithm for a class of Markov decision processes that includes the admission-control problem.

2. Submodularity and monotonic policies. We see from (2) that our maximization problem is of the form

$$f(i) = \max_a g(i,a) , \quad (6)$$

where in this case $g(i,a) = a r + V(i+a)$. The theory of submodular (and supermodular) functions (Topkis(1978)) provides a set of tools for showing monotonicity of the optimizing action in problems of this form. A function $g(i,a)$ is called submodular (supermodular) in (i,a) if $g(i,a') - g(i,a)$ is non-increasing in i , for all $a' > a$. If $a(i)$ is defined as the (smallest) maximizing action in (6), then it is easy to see that $a(i)$ is non-increasing (non-decreasing) in i if $g(i,a)$ is submodular (supermodular). (Symmetric statements hold for minimization problems.) In the present problem, submodularity of $g(i,a) = a r + V(i+a)$ follows directly from concavity of $V(i)$. This is often the case in queueing-control problems.

3. Extensions of the inductive approach. Variants of the inductive approach can be used to prove monotonicity of an optimal admission-control policy for systems with multiple servers or a state-dependent service mechanism, non-linear (convex) holding cost rate $h(i)$, and/or random rewards (Lippman and Stidham(1977)), with a general interarrival-time distribution, batch

arrivals with random batch size, or mixed-Erlang service-time distribution (Stidham(1978), Langen(1982)), with random environments, including dependent interarrival times, fixed time horizons, non-stationary arrival process, multiple job types, or partially observable processes (Helm and Waldmann(1983)), with general input and output process and continuous state variable (Johansen and Stidham(1980)), and with charging of rewards and costs at departure (rather than arrival) instants (Johansen and Stidham(1984)). For more details about other these and other single-facility admission-control models, see, e.g., Stidham and Prabhu(1974), Johansen and Stidham(1980), Stidham(1984).

2. Service Control in a Single-Server Queue

Our next model is for a single M/M/1 queue with control of service. The model is illustrated below in Figure 5. As we shall see, it is in some sense "dual" to the admission-control model of the previous section, and as such differs superficially from service-rate control models in the literature (Crabill(1972),(1974), Sabeti(1973), Lippman(1975), Jo(1982)). The formulations can be shown to be equivalent, however, by a simple transformation (see Remark 4 below).

Figure 5. M/M/1 Queue with Control of Service.

Once again jobs arrive from a Poisson process with mean arrival rate λ , but now all jobs enter the system. An entering job goes immediately into service if the server is free;

otherwise, it joins the (infinite-capacity) queue. The single server performs (potential) services according to an exponential distribution with mean service rate μ . Control is exercised by "accepting" or "rejecting" potential service completions when they occur. If a potential service completion is accepted, then a customer departs from the system and a cost $c \geq 0$ is incurred. If it is rejected then no cost is incurred and the system state remains unchanged. Alternatively, in the case of rejection we can think of the customer in service being "fed back" to receive another exponentially distributed service, as illustrated in Figure 5. As before, there is a holding cost h per unit time per job in the system. Future costs are continuously discounted at rate $\alpha > 0$ and the objective is to minimize the total expected discounted cost over an infinite horizon.

Define $V(i) :=$ minimum total expected discounted cost over an infinite horizon, starting from state i . Then this optimal value function satisfies the optimality equation ($i \geq 0$):

$$V(i) = [h \cdot i + \lambda V(i+1) + \mu \min(c + V(i-1), V(i))] / (\lambda + \mu + \alpha), \quad (7)$$

where $V(-1) = V(0)$. (Equivalently, assume that all potential service completions in state 0 are rejected.) The "duality" with equation (1) for the admission-control problem is obvious. The arguments for the validity of (7) -- both rigorous and heuristic -- parallel those for (1). An alternate version of the optimality equation, equivalent to (7), is

$$V(i) = [h \cdot i + \lambda V(i+1) + \mu V(i)]$$

$$+ \mu \min(c - [V(i) - V(i-1)], 0)] / (\lambda + \mu + \alpha), \quad (8)$$

from which it follows that an optimal service policy will be monotonic if $V(i) - V(i-1)$ is non-decreasing in i , that is, if $V(i)$ is convex. In this case, the optimal policy will be characterized by a critical number $i^* := \max\{i: V(i) - V(i-1) < c\}$. The quantity $V(i) - V(i-1)$ is the benefit (in terms of expected discounted future cost savings) of a service completion in state i . We should accept a potential service completion if and only if this benefit is at least as great as the service cost c , that is, if and only if $i > i^*$.

*Our convention is to accept a service completion in case cost exactly equals benefit.

Just as in the admission-control model, one can prove that $V(i)$ is convex by value iteration, and in the process also prove optimality of a monotonic policy for the n -stage problem. The key step is to show that convexity of a function $g(i)$ implies convexity of the function

$$f(i) := \min(c + g(i-1), g(i)) \quad (9)$$

The proof exactly parallels the proof of concavity of $f(i)$ when $f(i)$ is defined by (5) with $g(i)$ concave.

Remarks.

3. *Optimality of a full-service policy.* Under an optimal policy with critical number i^* , the states $i = 0, 1, \dots, i^*-1$ are transient. In other words, after the system first reaches state i^* , there will always be at least i^* jobs present. It may

seem strange that an optimal policy might not serve unless there are at least a certain minimum number of customers are present. In fact, when the criterion is long-run average return, Sobel(1982) has shown -- by a different approach and for much more general systems than the one under consideration here -- that $i^* = 1$. In other words, a "full-service" policy is optimal: service should take place whenever at least one job is in the system. But for discounted problems it is possible to have $i^* > 1$: if the discount rate is large enough, the savings in future holding costs, after discounting, may not be large enough in some states to offset the service cost, which is incurred now.

4. Control of service rate with one or more feasible values. In most of the literature on service control, the decision maker has the option of selecting, at each point in time, a service rate γ from a feasible set A , which may be a finite, countably infinite, or uncountable set (e.g., an interval $[0, \mu]$). When service rate γ is in effect, a cost is incurred at rate $c(\gamma)$ per unit time. (See, e.g., Crabill(1972),(1974), Sabetti(1973), Lippman(1975), Jo(1982).) By contrast, in the present model potential services take place at the constant rate μ and can be accepted, at a lump-sum cost c , or rejected, at no cost. There is an equivalence between the two types of model, which may be seen as follows.

First, our model is equivalent to one in which the decision maker must continuously choose between serving at rate μ , incurring service cost continuously at rate $c(\mu) := c$, or

serving at rate 0 (i.e., not serving), incurring service cost at rate 0. For, with $c(\mu)$ defined in this way, the optimality equation (7) can be rewritten as

$$V(i) = [h \cdot i + \lambda V(i+1) + \min\{c(\mu) + \mu V(i-1), \mu V(i)\}] / (\lambda + \mu + \alpha), \quad (10)$$

which is the optimality equation for the latter problem.*

 *As in the admission-control problem, we have uniformized the transitions by imagining that, even when the service rate is 0, the server continues to perform "fictitious" services at rate μ . These fictitious services have no effect on the state or costs, so that the resulting Markov decision process is equivalent to one in which only the "real" transitions (arrivals and service completions) are considered. (See Lippman(1975), Serfozo(1979).)

To see how the case of two or more feasible service rates can be handled, suppose our accept/reject model is modified as follows. There are k independent, parallel servers, the j -th of which serves according to an exponential distribution with mean rate γ_j , where $\sum_{j=1}^k \gamma_j = \mu$. Potential service completions by server j ($j = 1, 2, \dots, k$) can be accepted, at a cost c_j , or rejected, at 0 cost. Assume the c_j 's are non-decreasing in j . Then the term in the optimality equation (1) that involves minimization will be replaced by

$$\sum_{j=1}^k \gamma_j \cdot \min\{c_j + V(i-1), V(i)\}.$$

An optimal policy will accept service completions in state i from servers $j = 1, 2, \dots, j^*(i)$, and reject those from servers $j = j^*(i)+1, \dots, k$, where $j^*(i) := \max\{j: V(i) - V(i-1) \geq c_j\}$. The inductive proof that $V(i)$ is convex goes through without change. Thus the $j^*(i)$'s are non-decreasing in i : the more

jobs in the system, the more servers should be "on". This result, which is of interest in its own right, can be used to establish optimality of a monotonic policy for a single-server system in which the server can be operated at any rate $\gamma \in [0, \mu]$, at a cost per unit time $c(\gamma)$, where $c(\gamma)$ is a convex, non-decreasing function, with $c(0) = 0$. One simply approximates $c(\gamma)$ by a piecewise-linear function over k sub-intervals of length $\gamma_1, \dots, \gamma_k$, with $\gamma_1 + \dots + \gamma_k = \mu$ and $c_j := [c(\sum_{l=1}^j \gamma_l) - c(\sum_{l=1}^{j-1} \gamma_l)] / \gamma_j$ as the slope over the j -th subinterval. Convexity of $c(\gamma)$ implies that the c_j are non-decreasing in j , so that the above multi-server model applies. It follows that the optimal service rate to use in state i equals $\gamma_1 + \dots + \gamma_{j^*(i)}$, which is non-decreasing in i since $j^*(i)$ is non-decreasing in i .

The assumption that $c(\gamma)$ is convex is not restrictive, since any non-convex $c(\gamma)$ can be replaced by its lower convex envelope without affecting optimal policies. See, e.g., Crabill(1972), Jo(1982) for details. Intuitively, the reason is that any rate γ that belongs to an interval where $c(\gamma)$ is non-convex can be achieved at a lower cost by mixing two rates, one below and the other above γ .

5. Extensions and generalizations. Optimality of a monotonic policy can be proved by extensions of the above inductive analysis for systems with state-dependent arrival rates (Crabill(1972)), non-linear (convex) holding costs (Lippman(1975)), general service-time distribution with work-in-

system as the state variable (Mitchell(1970), Doshi(1976)), and phase-type service-time distributions (Jo and Stidham(1983)). Papers treating the service-control problem by other techniques include Schassberger(1976), Gallisch(1978), and Stidham and Weber(1984).

6. *Switching costs and hysteretic policies.* In some problems there may be a lump-sum cost associated with switching the service rate from one value to another, in addition to the cost rates associated with serving at various rates. (See Lu and Serfozo(1981) and the references cited therein.) This switching cost, for example, could be proportional to the difference between the old and new service rates. The optimal service rate γ to select at an observation point now depends on the rate ν currently in use as well as the number of jobs i , so that the appropriate state variable is (i, ν) . An inductive analysis can be used to show that an optimal policy is hysteretic, which means that it is characterized by a sequence of pairs of control limits, $((\underline{\nu}_i, \bar{\nu}_i), i=0,1,2,\dots)$, such that for each state (i, ν) the service rate should be (1) adjusted upward to $\underline{\nu}_i$, if $\nu < \underline{\nu}_i$; (2) adjusted downward to $\bar{\nu}_i$, if $\nu > \bar{\nu}_i$; and (3) left unchanged if $\underline{\nu}_i \leq \nu \leq \bar{\nu}_i$.

3. Control of Arrivals and Services in Cycles and Series of Queues

Our first model (Weber and Stidham(1983)) for control of a network of queues is for a cycle of m queues, in which a job that completes service at node (queue) i goes to node $i+1$.

(We identify node $m+1$ as node 1.) The system under consideration is illustrated in Figure 6. Jobs from outside the system enter node i at mean rate λ_i according to a Poisson process,

Figure 6. Cycle of Queues with Control of Service.

which is not subject to control. There is a single exponential server at node i who performs potential services at mean rate μ_i . A potential service completion may be accepted, at a cost c_i (which may be negative), or rejected, at 0 cost.* The

*Cf. the single-facility service-control model in Section 2. Continuous control of service rates and more than one feasible rate can be handled by the same transformations and extensions as used there. See Remark 4.

number of jobs in node i is denoted by x_i and a state of the system by the vector $x = (x_1, \dots, x_m)$, with $x_i \geq 0$, $i = 1, \dots, m$. While in state x , the system incurs holding cost per unit time $h(x) = \sum_i h_i(x_i)$, where each function $h_i(x_i)$ is non-negative and convex (but not necessarily non-decreasing). Future costs are continuously discounted at rate $\alpha > 0$ and the objective is to minimize the total expected discounted cost over an infinite horizon.

The two types of state transitions will be denoted

$$x \xrightarrow{A} x := x + e_i,$$

corresponding to an arrival at node i , and

$$x \xrightarrow{T} x := x - e_i + e_{i+1},$$

corresponding to an accepted service completion at node i and resulting transfer to node $i+1$. (Here e_i denotes an m -vector with 0's in every component except the i -th, which equals 1.) The system is observed at every arrival and potential service completion, so that observation points occur exponentially at rate $\Lambda := \sum_i (\lambda_i + \mu_i)$. Define $V(x) :=$ minimum total expected discounted cost over an infinite horizon, starting from state x . Then $V(x)$ satisfies the optimality equation

$$V(x) = [h(x) + \sum_{i=1}^m \lambda_i V(A_i x) + \sum_{i=1}^m \mu_i \min(V(x), c_i + V(T_i x))] / (\Lambda + \alpha), \quad (11)$$

where it is understood that the minimization operator selects $V(x)$ if $x_i = 0$. (The arguments for the validity of (11) parallel those for the single-facility model of Section 2.) We can rewrite (11) in the equivalent version

$$V(x) = [h(x) + \sum_{i=1}^m \lambda_i V(A_i x) + \sum_{i=1}^m \mu_i V(x) + \sum_{i=1}^m \mu_i \min(0, c_i - [V(x) - V(T_i x)])] / (\Lambda + \alpha). \quad (12)$$

The quantity $V(x) - V(T_i x)$ can be interpreted as the benefit of a service completion at node i in state x . We should accept a potential service completion at node i if and only if this benefit is at least as great as the service cost c_i .

Weber and Stidham (1983) showed that an optimal policy has the following property (analogous to simple monotonicity in single-facility problems):

After a service completion at node i (i.e., after a job is transferred from node i to node $i+1$), the optimal service rate at node i does not increase and the optimal service rates at other nodes do not decrease.

To establish this property, it suffices to show that the benefit, $V(x) - V(T_i x)$, of transferring a job from node i to node $i+1$ does not increase as another job is transferred from i to $i+1$, and does not decrease as another job is transferred from node j to node $j+1$, $j \neq i$. In other words,

$$V(x) - V(T_i x) \geq V(T_i x) - V(T_i T_i x), \quad (13)$$

$$V(x) - V(T_i x) \leq V(T_j x) - V(T_j T_i x), \quad j \neq i. \quad (14)$$

It can be shown (by moving one job all the way around the cycle) that (14) implies (13), so it suffices to prove (14).

We call a function $V(x)$ satisfying (14) multimodular, following Hajek(1983), who introduced the concept in a different context. In a sense, multimodularity is a multi-dimensional analogue of submodularity. Weber and Stidham(1983) prove that the optimal value function $V(x)$ is multimodular by a value-iteration induction, the key step of which (cf. the single-facility models) is to show that multimodularity is preserved by transformations of the form

$$f(x) = \min_k \{c_k + g(T_k x), g(x)\}.$$

Applications: Series of Queues with Control of Arrivals and/or Services.

These results can be applied to a series of queues ($i=1,2,\dots,m-1$), with control of the (Poisson) arrival process at the first node, by adding to the series a dummy node m , with no holding cost and an infinite supply of jobs, and letting that node receive all output from node $m-1$ and generate all input to node 1. Accepting or rejecting service completions at the dummy node corresponds to accepting or rejecting arrivals to the first node in the series. The service cost c_m at the dummy node is the negative of the reward r earned when an arriving job is accepted at node 1.

The monotonicity result referred to above implies that the benefit of accepting an arriving job does not decrease as another job is transferred from any node j in the series to node $j+1$, or (by combining a sequence of such moves) as a job is removed from any node j in the series. Thus an optimal admission-control policy will be more likely to accept if either of these two types of state change is made, which generalizes the result in Section 1. Although the present model implicitly assumes that the service rate at each of the nodes in series is also controllable, the results apply to a system with fixed service rates, as long as the marginal holding costs do not increase from node i to node $i+1$, $i = 1,2,\dots,m-1$. For in this case the problem with fixed service rates is equivalent to one in which service completions can be accepted or rejected but there is no cost of accepting a service completion; in the latter problem it

will always be optimal to accept all service completions, since they move a job to a cheaper node at no cost. Note that this ordering of the marginal holding costs implies that each holding cost function $h_i(x_i)$ is non-decreasing, since the marginal holding cost at node m is identically zero. See Weber and Stidham(1983) for further discussion.

The results of Weber and Stidham(1983) may be compared to those of Lazar(1983), who also analyzes control of the arrival rate to the first queue in a series of queues. Lazar studies a steady-state version of the problem, in which the objective is to maximize expected steady-state throughput, subject to a constraint on the total expected response time (time to pass through all nodes) of a job. By a Lagrange-multipliers argument, this problem can be seen to be equivalent to the problem of choosing an arrival rate to the first node that maximizes a weighted sum of the arrival rate (i.e., the throughput of the system) and minus the steady-state expected number of customers in the system. The latter problem is clearly equivalent to the problem of maximizing the long-run average net benefit in a system with fixed reward per admitted customer and linear holding costs at each queue, with the same holding-cost coefficient: a problem that belongs to the class considered in Weber and Stidham(1983).

Lazar's analysis shows that an optimal control policy has a critical-number form with respect to the total number of jobs in the series: "end-to-end" control is optimal. This result may seem to contradict those in Weber and Stidham(1983), in which optimal policies can (and generally do) have a more general

structure. But in fact there is no contradiction, since Lazar derives his result by first considering the single-facility Norton equivalent to the series, for which (of course) an optimal policy is a function of the total number in the system. Therefore, by the way in which Lazar has formulated his problem, he has from the beginning restricted attention to policies based on the total number of customers in the series.

Other Network-Control Models.

Davis(1977) considered two exponential servers (with mean rates μ_1 and μ_2) in parallel, each with its own queue, and a renewal arrival process -- that is, i.i.d. interarrival times distributed as a random variable T . The system controller may reject ($a=0$) an arriving job, admit it to queue 1 ($a=1$), or admit it to queue 2 ($a=2$), based on the state $x = (x_1, x_2)$ at the instant of arrival, where $x_j :=$ number of jobs at queue j (including the one in service, if any), $j = 1, 2$. Figure 7 illustrates the model. An admitted customer generates a (deter-

Figure 7. Control of Admission and Routing to Two Parallel Queues

ministic) reward r . There is a holding cost rate $h_j(x_j)$ per unit time while x_j jobs are at queue j , where $h_j(\cdot)$ is a convex, non-decreasing function, $j = 1, 2$.

An inductive proof based on value iteration shows that an optimal admission/routing policy $\{a(x)\}$ is monotonic for this problem, in the sense that $a(x) = 0$ implies $a(x+e_j) = 0$, $j =$

1,2 ; in other words, the rejection region $R := \{x: a(x) = 0\}$ is an increasing set. The induction also shows that the "switching curve" is monotonic: if admitting to queue 2 (queue 1) is preferable to admitting to queue 1 (queue 2) in state x , then it will remain so in state $x + e_1$ ($x + e_2$). Finally, an additional property of the rejection region is demonstrated by the induction: as we move closer to the switching curve, we are more likely to reject a job. To illustrate the application of this property, note that each of the rejection regions illustrated in Figure 8 below (for the case of two symmetric queues) is an increasing set, but 8(a) and 8(b) violate this property and thus cannot be rejection regions for an optimal policy. All

Figure 8. Examples of Increasing Sets for Parallel-Queue Problem

these properties of an optimal policy follow from verifying that the optimal value function is concave in each argument, submodular, and satisfies a third condition. The three conditions taken together constitute the analogue of multimodularity for maximization problems in two dimensions. Control of routing with parallel queues was also considered by Farrell(1976), Winston(1977), Weber(1978), and Ephremides, Varaiya, and Walrand(1979) among others.

Ghoneim(1980) (see also Ghoneim and Stidham(1983)) studied two exponential servers in series (with mean service rates μ_1 and μ_2), each with an infinite-capacity queue. Arrivals to queue j are from a Poisson process with mean rate λ_j , $j = 1,2$. Jobs arriving to queue 1 must go on to queue 2 after

finishing service at server 1 . Jobs arriving to queue 2 leave the system after finishing service at server 2 . (See Figure 9.) The model thus describes, for example, a simple communication system consisting of two channels in series with a combination of local and long-distance traffic.

Figure 9. Control of Admission to Two Queues in Series.

With the same reward and cost structure as for the parallel-queue model just discussed, an induction based on value iteration establishes that the same three properties hold for the optimal value function. Thus, for example, the optimal rejection region for jobs arriving to queue 1 is an increasing set. These properties also rule out certain increasing sets as candidates for the optimal rejection region, namely those whose boundaries have horizontal segments of length greater than one.

A typical optimal rejection region (from a numerical example) has the shape shown in Figure 10. (This shape for the

Figure 10. Typical Optimal Rejection Region for Two Queues in Series

rejection region is characteristic of both discounted and undiscounted problems.) It may be instructive to compare this rejection region to that implied by one of the flow-control rules proposed in the communications literature. "End-to-end" control (Lazar(1983)) suggests putting an upper bound, k , on the total number of jobs (messages) in a particular path in the network. The corresponding rejection region has a straight-line boundary, $x_1 + x_2 = k$. By contrast the boundary of an optimal rejection

region, as illustrated in Figure 10, is non-linear and non-symmetric. Numerical examples (cf. Abdel-Gawad(1984)) have shown that an optimal policy can yield benefits up to 15% larger than those from the best end-to-end control. Implementing an optimal control requires that the system controller keep track of the number of jobs at each node (queue) in a path, rather than just the total number. So additional bookkeeping overhead would be required.

Hajek(1982) has considered a general two-node model that incorporates many of the features of both the parallel and series queue models (but not the option of accepting or rejecting arriving jobs). In Hajek's model, queues 1 and 2 receive Poisson arrivals at rates λ_1 and λ_2 , respectively. A third stream of Poisson arrivals at rate λ can be routed to either queue. The stations have fixed exponential servers with rates μ_1 and μ_2 and a third exponential server with rate μ that can be assigned to either queue; jobs whose service is completed by these servers leave the system. There are two additional exponential servers, with rates γ_{12} and γ_{21} , the first of which serves queue 1 and sends jobs to queue 2, the second of which serves queue 2 and sends jobs to queue 1. Service completions by these servers can be "accepted" or "rejected"; the jobs arriving at rate λ are to be routed to one or the other of the queues; and the server with rate μ is to be assigned to one or the other of the queues. All these decisions are to be made dynamically as a function of the number of jobs in the two queues. Hajek uses an inductive proof to establish the existence of a

monotonic switching curve, on which all these decisions can be based. His analysis accommodates convex holding costs at each queue and costs associated with each switching decision.

So far little progress has been made in characterizing or computing optimal control policies for more complicated networks than those we have discussed. As far as I know, the only successful analysis of a network with more than two nodes is that of Weber and Stidham(1983). An essential feature of their cycles/series model is the absence of branching or routing choices. As we have seen, both branching and routing choices have been successfully analyzed in the context of two-node problems. But attempts to extend these results to more than two nodes have failed. In particular, the three-node, series-parallel network discussed in the introduction (see Figure 3) apparently cannot be studied by the inductive approach. (See Abdel-Gawad(1984) for further discussion.)

References

- ABDEL-GAWAD, E.F. (1984) Optimal control of arrivals and routing in a network of queues. Ph.D. dissertation, Program in Operations Research, N.C. State University, Raleigh.
- ADLER, I., and NAOR, P. (1969) Social optimization versus self-optimization in waiting lines. Tech. Rpt. No. 4, Department of Operations Research, Stanford University.
- ALTIOK, T. (1982) The allocation of buffer capacities in stochastic production lines. Ph.D. dissertation, Department of Industrial Engineering, N.C. State University, Raleigh.
- ALTIOK, T., and STIDHAM, S. (1983) The allocation of interstage buffer capacities in production lines. *IIE Transactions* 15, 292-299.
- BALACHANDRAN, K., and SCHAEFER, M. (1979) Class dominance characteristics at a service facility. *Econometrica* 42, 515-519.
- BELLMAN, R. (1957) *Dynamic Programming*. Princeton: Princeton University Press.
- BENGTSSON, B. (1982) On some control problems for queues. Linköping Studies in Science and Technology, Dissertation No. 87, Department of Electrical Engineering, Linköping University, Sweden.
- BERTSEKAS, D. (1976) *Dynamic Programming and Stochastic Control*. New York: Academic Press.
- BUZACOTT, J., and YAO, D. (1982) Flexible manufacturing Systems: A Review of Models. Working Paper No. 7, Department of Industrial Engineering, University of Toronto.
- CRABILL, T. (1972) Optimal control of a service facility with variable exponential service time and constant arrival rate. *Management Sci.* 18, 560-566.
- CRABILL, T. (1974) Optimal control of a maintenance system with variable service rates. *Operations Res.* 22, 736-745.
- CRABILL, T., GROSS, D., and MAGAZINE, M. (1977) A classified bibliography of research on optimal design and control of queues. *Operations Res.* 25, 219-232.
- DAVIS, E. (1977) Optimal control of arrivals to a two-server queueing system with separate queues. Ph.D. dissertation, Program in Operations Research, N.C. State University, Raleigh.
- DOSHI, B. (1977) Continuous-time control of the arrival process in

- an M/G/1 queue. *Stoch. Process Appl.* 5, 265-284.
- DOSHI, B. (1978) Optimal control of the service rate in an M/G/1 queueing system. *Adv. Appl. Prob.* 10, 682-701.
- EPHREMIDES, A., VARAIYA, P., and WALRAND, J. (1980) A simple dynamic routing problem. *IEEE Trans. Auto. Control* AC-25, 690-693.
- FARRELL, W. (1976) Optimal switching policies in a nonhomogeneous exponential queueing system. Ph.D. dissertation, Graduate School of Management, University of California, Los Angeles.
- GALLISCH, R. (1979) On monotone optimal policies in a queueing system of M/G/1 type with controllable service-time distribution. *Adv. Appl. Prob.* 11, 870-887.
- GHONEIM, H. (1980) Optimal control of arrivals to a network of two queues in series. Ph.D. dissertation, Program in Operations Research, N.C. State University, Raleigh.
- GHONEIM, H., and STIDHAM, S. (1983) Optimal control of arrivals to two queues in series. Submitted to *Europ. J. Operations Res.*
- GROSS, D., MILLER, D., and SOLAND, R. (1983) A closed-queueing-network model for multi-echelon repairable-item provisioning. *IEE Transactions* 15, 344-352.
- HAJEK, B. (1982) Optimal control of two interacting service stations. To appear in *IEEE Trans. Auto. Control*.
- HAJEK, B. (1983) Extremal splitting of a point process. To appear in *Math. of Operations Res.*
- HELM, W., and WALDMANN, K.-H. (1983) Optimal control of arrivals to multiserver queues in a random environment. To appear in *Adv. Appl. Prob.*
- HO, Y., and CAO, X. (1983) Perturbation analysis and optimization of queueing networks. *J. Optimization Theory and Appl.* 40.
- JACKSON, J.R. (1963) Jobshop-like queueing systems. *Management Sci.* 10, 131-142.
- JO, K. (1982) Optimal control of service rates in queues and networks of queues. Ph.D. dissertation, Program in Operations Research, N.C. State University, Raleigh.
- JO, K. (1983) Optimal service-rate control of exponential systems. *J. Operat. Res. Soc. Japan* 24, 147-165.
- JO, K., and STIDHAM, S. (1983) Optimal service-rate control of M/G/1 queueing systems using phase methods. *Adv. Appl. Prob.* 15, 616-637.

- JOHANSEN, S.G., and STIDHAM, S. (1980) Control of arrivals to a stochastic input-output system. *Adv. Appl. Prob.* 12, 972-999.
- JOHANSEN, S.G., and STIDHAM, S. (1984) The optimal pricing of admission to a GI/M/c queue. Department of Operations Research, Aarhus University, Denmark.
- KLEINROCK, L. (1976) *Queueing Systems*, Vols. I and II. New York: John Wiley and Sons.
- KNUDSEN, N.C. (1972) Individual and social optimization in a multi-server queue with a general cost-benefit structure. *Econometrica* 40, 515-528.
- KNUDSEN, N.C., and STIDHAM, S. (1976) Individual and social optimization in a birth-death congestion system with a general cost-benefit structure. Tech. Rpt. No. 76-8, Department of Industrial Engineering, N.C. State University, Raleigh.
- LANGEN, H.-J. (1982) Applying the method of phases in the optimization of queueing systems. *Adv. Appl. Prob.* 14, 122-142.
- LAZAR, A. (1983) Optimal flow control of a class of queueing networks in equilibrium. *IEEE Trans. Auto. Control* AC-28, 1001-1007.
- LIPPMAN, S. (1975) Applying a new device in the optimization of exponential queueing systems. *Operations Res.* 23, 687-710.
- LIPPMAN, S., and STIDHAM, S. (1977) Individual versus social optimization in exponential congestion systems. *Operations Res.* 25, 233-247.
- LU, F., and SERFOZO, R. (1981) M/M/1 queueing decision processes with monotone hysteretic optimal policies. To appear in *Operations Res.*
- MITCHELL, W. (1973) Optimal service-rate selection in an M/G/1 queue. *Siam J. Appl. Math.* 24, 19-35.
- NAOR, P. (1969) On the regulation of queue size by levying tolls. *Econometrica* 37, 15-24.
- NUNEN, J. van, and PUTERMAN, M. (1980) On solving G/M/s queueing control systems. *Operations Research Proceedings*. Berlin: Springer-Verlag, 483-490.
- NUNEN, J. van, and PUTERMAN, M. (1982) Computing optimal control limits for GI/M/s queueing systems with controlled arrivals. Submitted to *Management Science*.

- NUNEN, J. van, and WESSELS, J. (1979) Successive approximations for Markov decision processes and Markov games with unbounded rewards. *Math. Operationsforsch. Statist., Ser. Optimization* 10, 431-455.
- ROSS, S. (1970) *Applied Probability Models with Optimization Applications*. San Francisco: Holden-Day.
- RUE, R., and ROSENSHINE, M. (1981) Optimal control of entry of many classes of customers to an M/M/1 queue. *Naval Res. Logist. Quant.* 28, 489-495.
- SABETI, H. (1973) Optimal selection of service rates in queueing with different costs. *J. Operat. Res. Soc. Japan* 14, 15-35.
- SCHAL, M. (1975) Conditions for optimality in dynamic programming and for the limit of n-stage optimal policies to be optimal. *Z. Wahrscheinlichkeitstheorie verw. Geb.* 32, 179-196.
- SCHASSBERGER, R. (1975) A note on optimal service selection in a single-server queue. *Management Sci.* 21, 1326-1331.
- SERFOZO, R. (1979) An equivalence between discrete- and continuous-time Markov decision processes. *Operations Res.* 22, 616-620.
- SERFOZO, R. (1981) Optimal control of random walks, birth and death processes, and queues. *Adv. Appl. Prob.* 13, 61-83.
- SIMONOVITS, A. (1976) Self- and social optimization in queues. *Studia Scient. Math. Hungarica* 11, 131-138.
- SOBEL, M. (1974) Optimal operation of queues. In *Mathematical Methods in Queueing Theory*, A.B. Clarke (ed.), Berlin: Springer-Verlag, 231-261.
- SOBEL, M. (1982) The optimality of full-service policies. *Operations Res.* 30, 636-649.
- STIDHAM, S. (1978) Socially and individually optimal control of arrivals to a GI/M/1 queue. *Management Sci.* 24, 1598-1610.
- STIDHAM, S. (1984) Optimal control of admission to a queueing system. Submitted to *IEEE Trans. Auto. Control*.
- STIDHAM, S., and PRABHU, N.U. (1974) Optimal control of queueing systems. In *Mathematical Methods in Queueing Theory*, A.B. Clarke (ed.), Berlin: Springer-Verlag, 263-294.
- STIDHAM, S., and WEBER, R. (1984) Monotonic and insensitive optimal policies for control of arrival and service rates in queues. Program in Operations Research, N.C. State University, Raleigh. Submitted for publication.
- TOPKIS, D. (1978) Minimizing a submodular function on a lattice.

Operations Res. 24, 305-321.

WAL, J. van der (1981) Stochastic Dynamic Programming. Mathematical Centre Tract 139, Amsterdam.

WEBER, R. (1978) On the optimal assignment of customers to parallel servers. J. Appl. Prob. 15, 406-413.

WEBER, R., and STIDHAM, S. (1983) Control of service rates in cycles and series of queues. Statistical Laboratory, University of Cambridge. Submitted for publication.

WIJNGAARD, J., and STIDHAM, S. (1983) Forward recursion for Markov decision processes with skip-free-to-the-right transition structure. Program in Operations Research, N.C. State University, Raleigh. Submitted for publication.

WINSTON, W. (1977) Optimality of the shortest-processing time discipline. J. Appl. Prob. 14, 181-189.

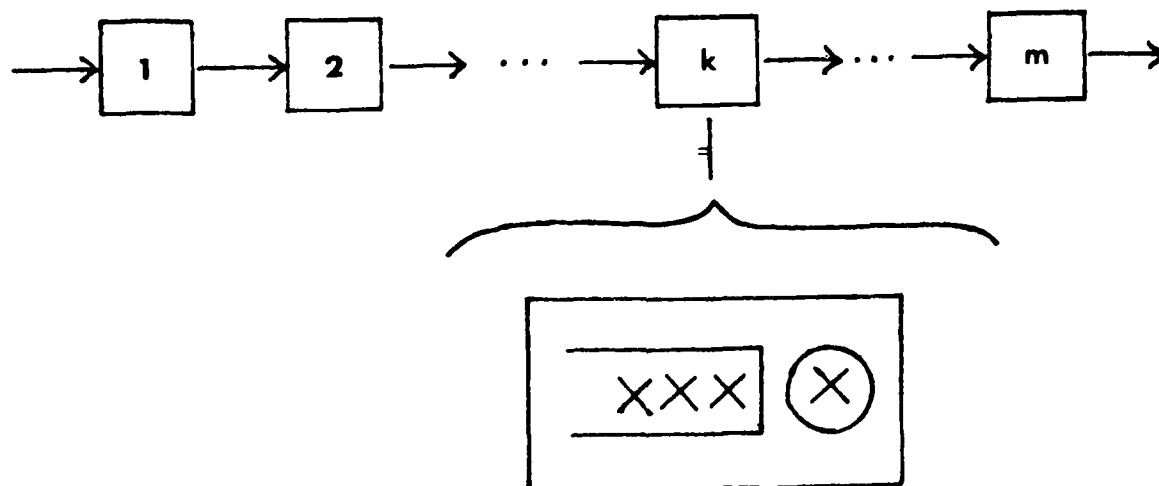


Figure 1. Queues in Series.

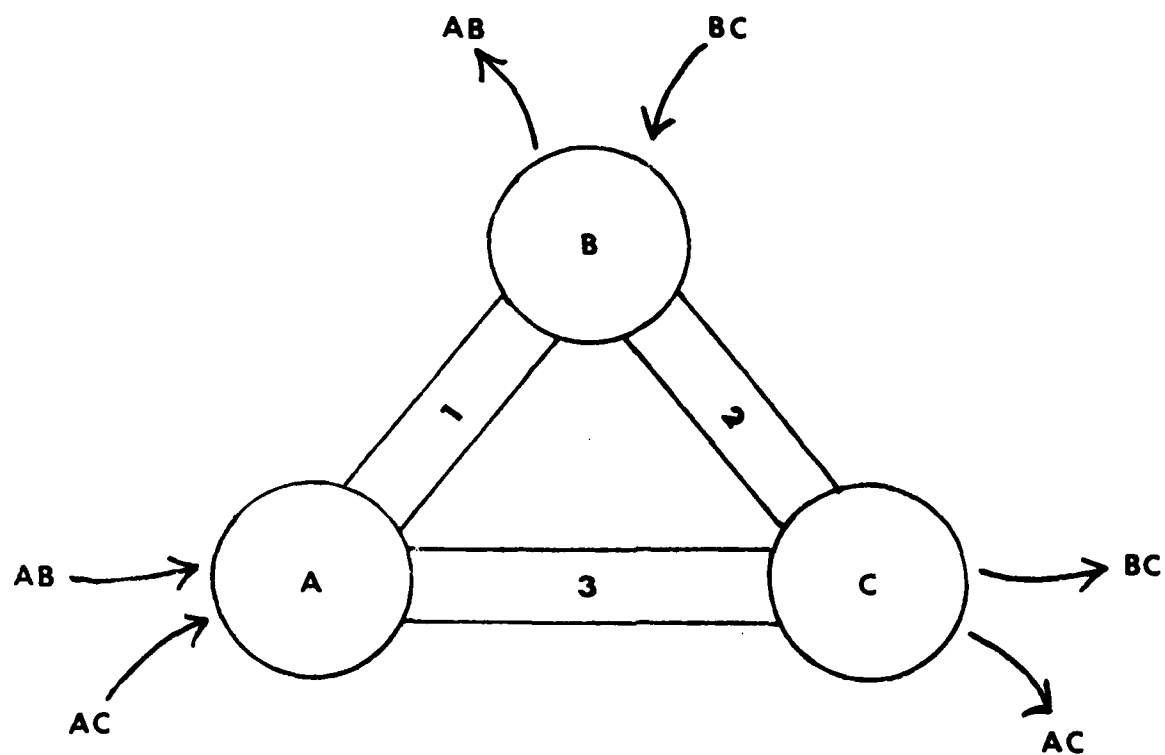


Figure 2. Simple Communication Example.

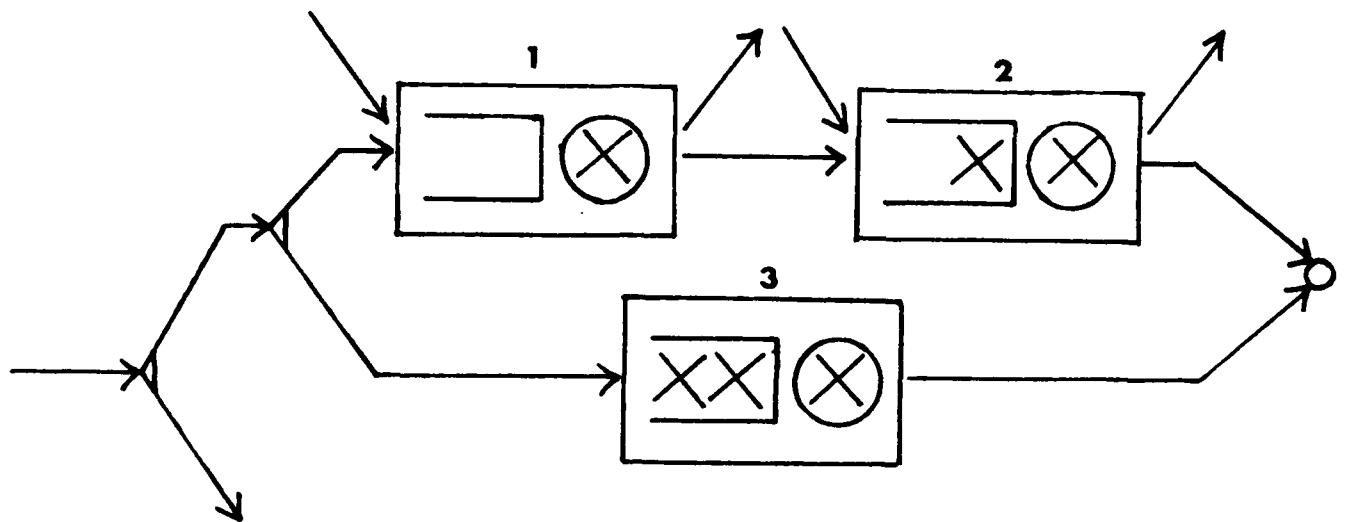


Figure 3. Queueing-Network Model for Communication System.

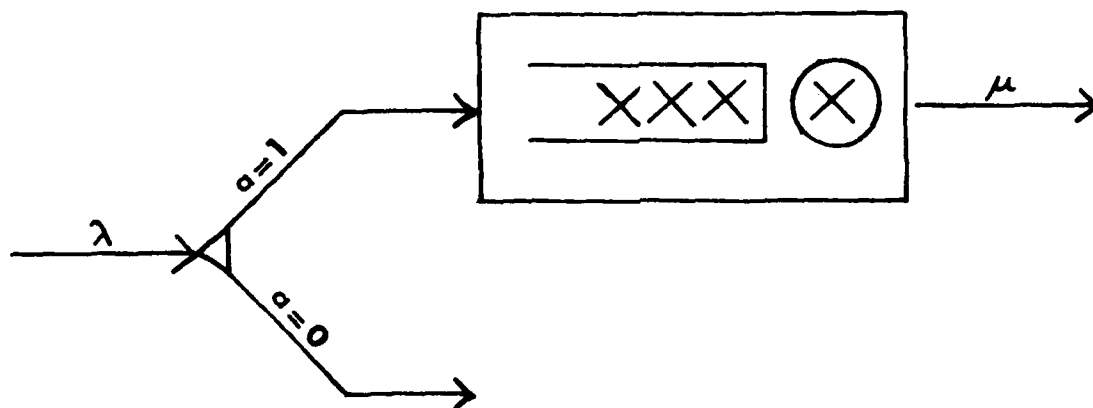


Figure 4. M/M/1 Queue with Control of Admission.

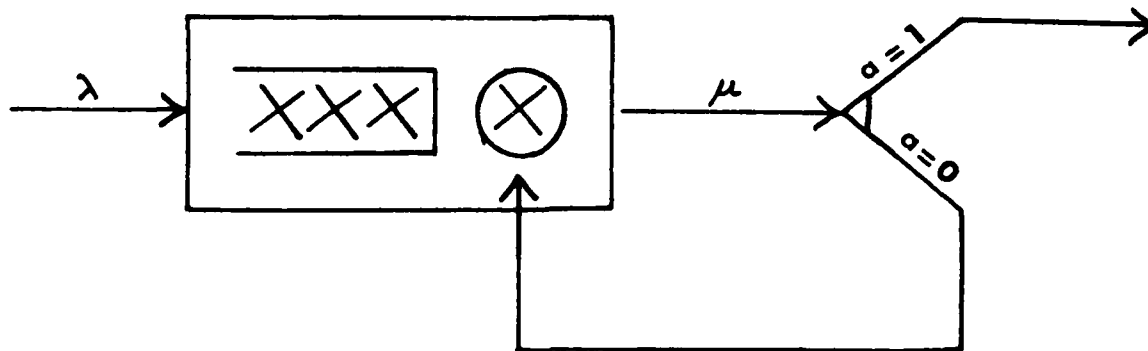


Figure 5. M/M/1 Queue with Control of Service.

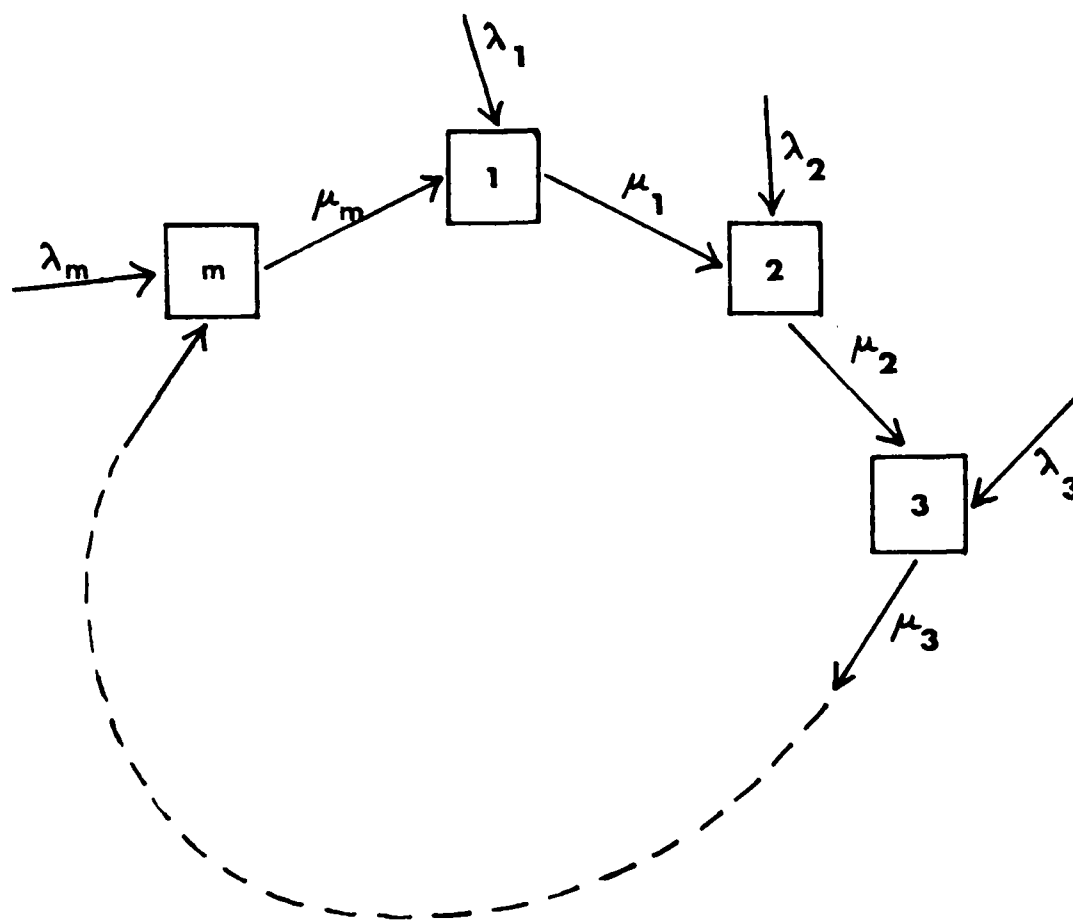


Figure 6. Cycle of Queues with Control of Service.

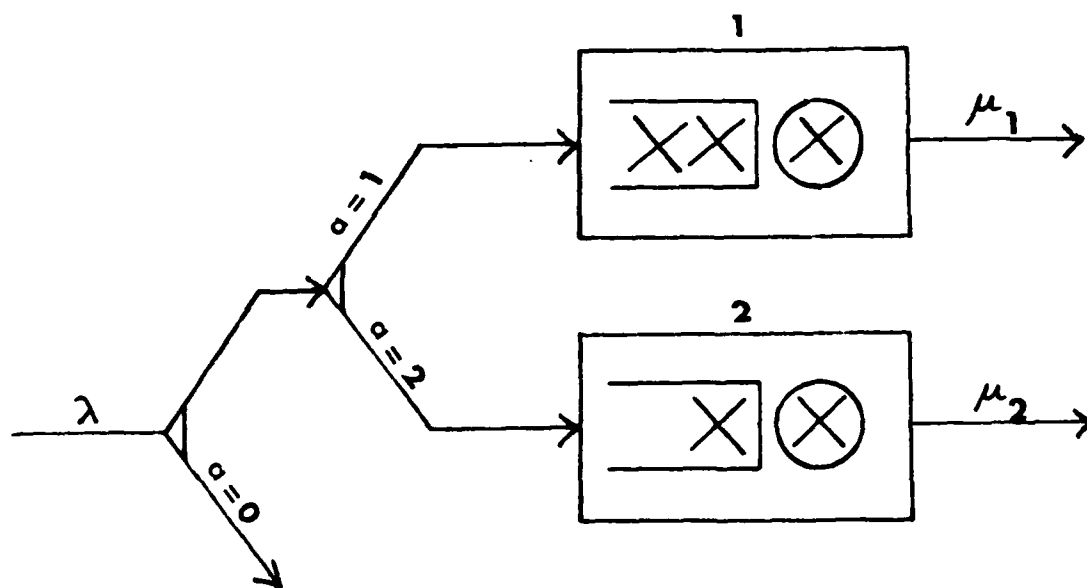
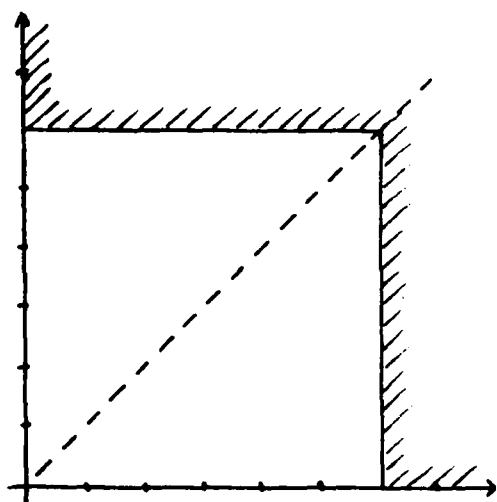
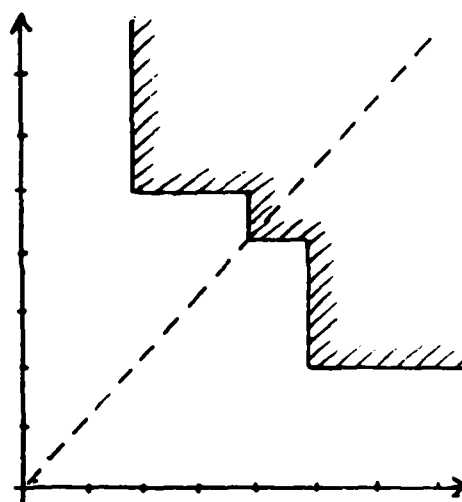


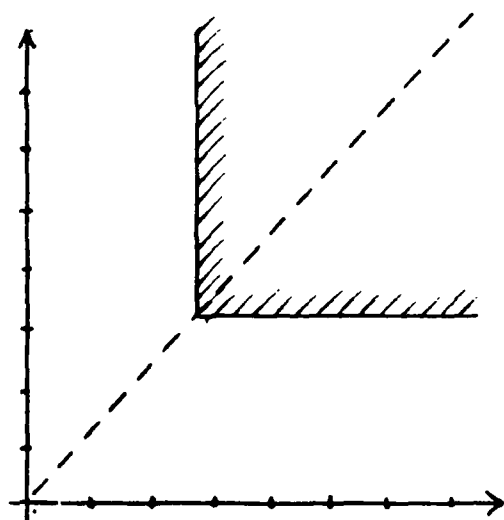
Figure 7. Control of Admission and Routing to Two Parallel Queues.



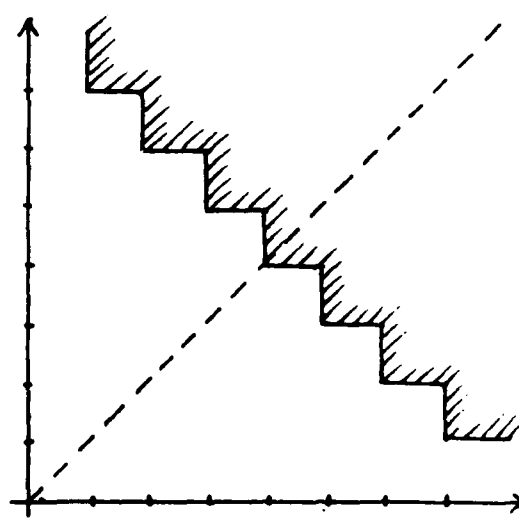
(a)



(b)



(c)



(d)

Figure 8. Examples of Increasing Sets for Parallel-Queue Problem.

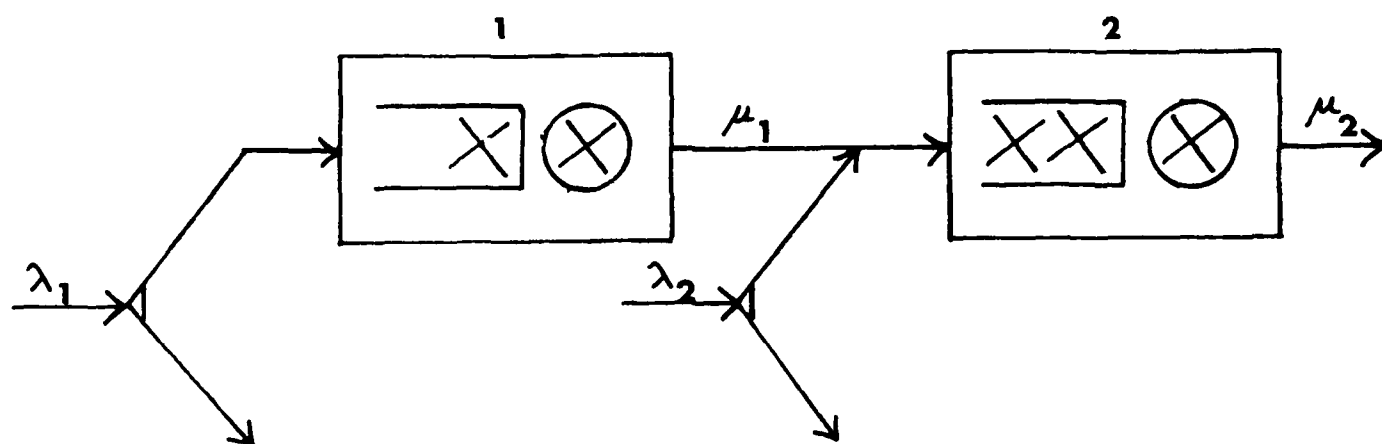


Figure 9. Control of Admission to Two Queues in Series.

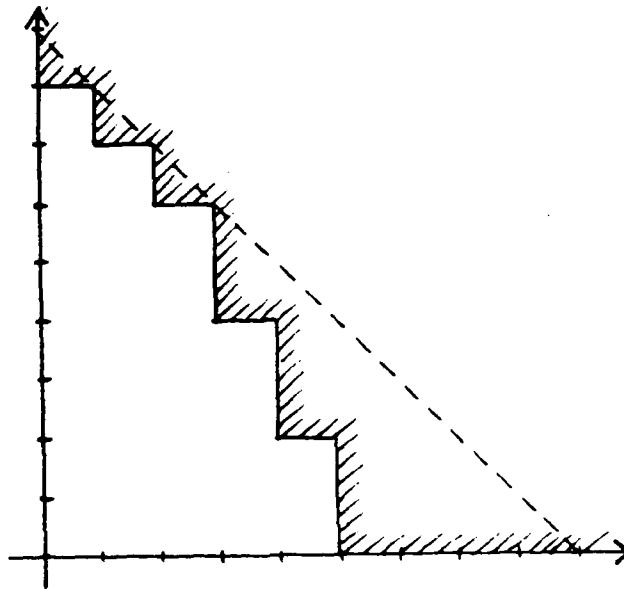


Figure 10. Typical Optimal Rejection Region for Two Queues in Series.

SOME RECENT ADVANCES IN ACTIVITY NETWORKS

S.E. Elmaghraby⁽¹⁾, B.M. Dodin⁽²⁾,
A.M. Salem⁽³⁾, and A.H. Colby⁽⁴⁾

OR REPORT NO. 198

MAY 1984

Paper prepared for presentation at the ARO Workshop on "Analytical and Computational Issues in Logistics R&D", Washington, D.C. May 7-9, 1984. This research was partially supported by ARO Contract No. DAAG29-79-C-0211.

- (1) N.C. State University, Graduate Program in Operations Research.
- (2) University of Wisconsin-Milwaukee, School of Business Administration.
- (3) Logistics Command, Egyptian Navy, Alexandria, Egypt.
- (4) N.C. State University and the U.S. Military Academy, West Point, NY.

SOME RECENT ADVANCES IN ACTIVITY NETWORKS

S. E. Elmaghraby
NCSU

INTRODUCTION

This paper is a subset of a much larger monograph on Progress in ANs[†] that is currently under preparation. Here, we limit ourselves to three specific areas: Project compression in DANs^{*}, statistical estimation in PANs[§], and reducibility of ANs. (For a more detailed exposition of the acronyms DANs and PANs, see the book by Elmaghraby [8].)

The CPM model [8] resolved the questions that first come to mind in DANs, whose analysis is, fortunately, of elementary nature. They enriched our vocabulary with such important concepts as: Critical path (CP), event "slack" and activity "float", earliest and latest node realization times; etc. The two major outstanding problems are: optimal project "compression", and optimal "resource allocation"; neither of which can be termed "elementary". Indeed, the latter problem, that of optimal resource allocation, is known to be NP-Complete and the achievement of the optimum is impractical except in the most trivial of networks. This, in turn, signifies that the problem is difficult to resolve by an "efficient" algorithm for any realistic AN; hence the rash of heuristic procedures that yield "good" answers.

[†] Activity Network

^{*} Deterministic Activity Network

[§] Probabilistic Activity Network

Fortunately, the problem of project compression is amenable to resolution. Its significance resides in the ability to specify the most efficient utilization of investments in the speeding up of a project. Alternatively, it serves to alert the manager to the range of requirements of additional investments should he wish to deviate appreciably from the "normal" flow of work in the project.

The problems of statistical estimation are concerned with the determination of probability distribution functions (pdf) of the time of realization of nodes (= events) when the durations of the activities are random variables (r.v.). In addition, a host of other issues are raised relative to the criticality of paths and activities, whose answer is difficult to compute, despite their theoretical simplicity. Approximations and bounding techniques are used to give the analyst the insight desired.

Interestingly enough, the above two classes of problems give rise to the third issue discussed in this report, viz., the reducibility of ANs.

One final remark. The three main sections of this report are almost independent. This may have introduced some redundancy, but should facilitate reading.

I. OPTIMAL PROJECT COMPRESSION

The mathematical statement of the problem runs as follows:

$$\text{minimize } \sum_{(ij) \in A} C_{ij}(y_{ij}) \quad (1)$$

such that the precedence constraints are respected, and the project is completed on or before time T_s . Let t_i denote the time of realization of node i . Then if activity $(ij) \in A$, with the arrow in the direction $i \rightarrow j$, we must impose the restriction

$$-t_i + t_j - y_{ij} \geq 0 \quad \forall (ij) \in A \quad (2)$$

The completion time requirement adds the constraint:

$$t_1 - t_n \geq -T_s \quad (3)$$

Here we assume that the "start node" is node 1 , and that the "terminal" node is n ; whence the set $N \equiv \{1, 2, \dots, n\}$. Finally, the activity duration y_{ij} is bound from below by a lower limit $\ell_{ij} \geq 0$, and from above by an upper limit $u_{ij} > \ell_{ij}$; i.e., $0 \leq \ell_{ij} \leq y_{ij} \leq u_{ij}$. (The only instance in which y_{ij} is permitted to be 0 is in the case of "dummy" activities; see ref. [8] for a detailed explanation of the utility of these activities.) It is more convenient to re-write this double inequality as

$$y_{ij} \geq \ell_{ij} \quad \text{and} \quad -y_{ij} \geq -u_{ij}; \quad \forall (ij) \in A \quad (4)$$

The mathematical program (1)-(4) has been extensively studied under the various manifestations of the individual activity time-cost function C_{ij} , (see Chapter 2 of ref. [8] for details). We devote the remainder of this section to the analysis of the case in which $C_{ij}(y_{ij})$ is convex decreasing.

In passing, we mention that, to the best of our knowledge, the first treatment of convex cost functions was by Jewel [14] in 1965. However, his motivation stemmed from PANs, where his objective was to balance the cost of project compression versus contract penalties and bids by competitors. In particular, he addressed the following question: A fixed project schedule must be determined now despite uncertainty in activity durations. Based upon the difference between the allotted time interval and the "free" time needed by the job, corrective action may have to be taken to stay within the fixed, predetermined, schedule. The problem is to determine that schedule that minimizes the expected amount of extra effort expended to stay on schedule.

He assumes that if the allotted time to an activity (ij) is $z_{ij}(= t_j - t_i)$, and if the activity (random) duration is Y_{ij} , then the cost function $g(z|Y)$ is convex in z for every realization y of Y_1 for all activities. In Figure 1, three sample curves are given, and all differ in the cost incurred if y is actually less than z : (I) represents resulting economy through, for instance, resource utilization elsewhere; (II) would obtain if the committed resources are irretrievable; and (III) represents the need to spend more effort because of, say, disposal activities.

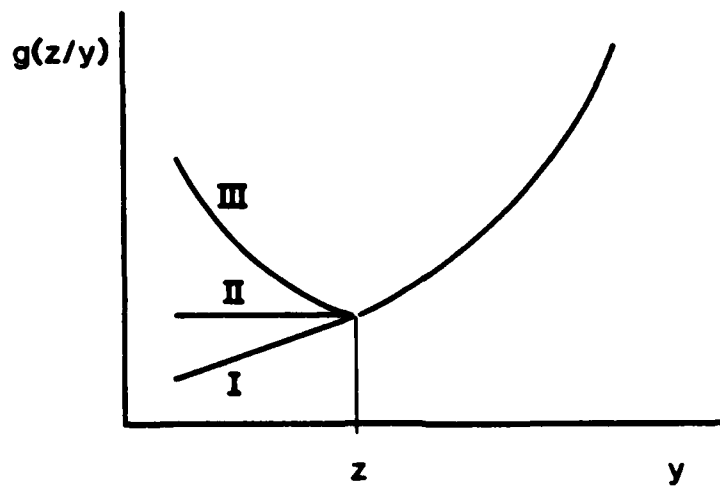


Figure 1. Three alternative curves of effort required as a function of actual completion time y .

Convexity of g in z for all y guarantees the convexity of the expected cost in the decision variable z , as well as the convexity of the sum of costs over all activities, i.e., the convexity of

$$\bar{g}(z) = \int_0^{\infty} g(z|y) dF(y)$$

where $F(y)$ is the pdf of y , and of

$$\bar{G}(Z) = \sum_{(ij)} \bar{g}_{ij}(z_{ij})$$

A typical curve of $\bar{g}(z)$ is shown in Figure 2.

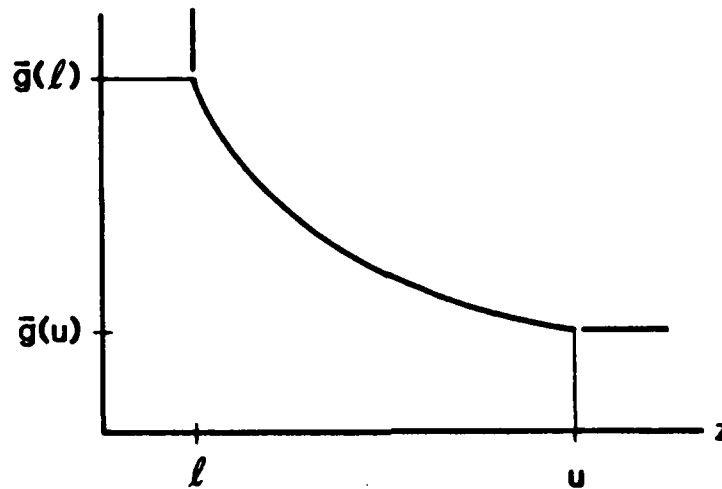


Figure 2. Resulting expected cost.

From this point onwards, the treatment is devoid of uncertainty, and is reduced to minimizing $\bar{G}(Z)$ subject to the usual precedence constraints. Jewell advocates approximating $\bar{g}(z)$ by linear segments, and then applying Fulkerson's algorithm. (In the example solved in the paper, the function $\bar{g}(z)$ was quadratic in z and he used a quadratic programming algorithm.) More elaborate approaches to this problem are the subject matter of the remainder of this section.

We start with the quadratic case, which can be resolved analytically, while the general case is approximated by linear splines.

As will become evident below, the conclusions are more transparent if we discuss two cases separately: The first assumes that the derivative dC/dy (sometimes also denoted by C') is continuous for $y \in [l, \infty)$; and the second accepts discontinuities at $l > 0$ and $u > l$.

Our discussion covers two trains of thought: The first is to achieve exact solutions, and the second is to approximate the optimum. As will be seen, each raises its own secondary problems.

1. EXACT SOLUTIONS

Case 1: Continuous Derivative

Since C is quadratic with continuous derivative in the interval $[l, \infty)$, we may assume it, without any loss of generality, to be of the form

$$C(y) = b + \beta(u-y)^2 \quad l \leq y \leq u \quad (5)$$

Note that C is tangent to the line $C(y) = b$ at $y = u$, where $C' = 0$; see Fig. 3. We may go one step further and simplify (5) to

$$C(x) = b + (\hat{u} - x)^2$$

by setting $\hat{u} = u \sqrt{\beta}$ and $x = y \sqrt{\beta}$, the "normalized" values of u and y , respectively. Henceforth, we drop the " $\hat{\cdot}$ " from the \hat{u} for the sake of simplicity, since the context reveals to which value of the upper bound reference is made. In general, analysis proceeds with the normalized variables $\{x_{ij}\}$ to the end, then it is translated into the original $\{y_{ij}\}$ -variables.

We introduce one mild assumption whose justification is easy to establish: T_g is such that no activity will be at its lower bound; i.e., at the optimum, $y_{ij} > l_{ij}$, $(ij) \in A$. (Note that if l_{ij} is small enough, this condition will be automatically satisfied.) We shall refer to it as Condition L.

In ref. [7] the following characterization of the optimum solution is given. Let the nodes of the network be realized at times

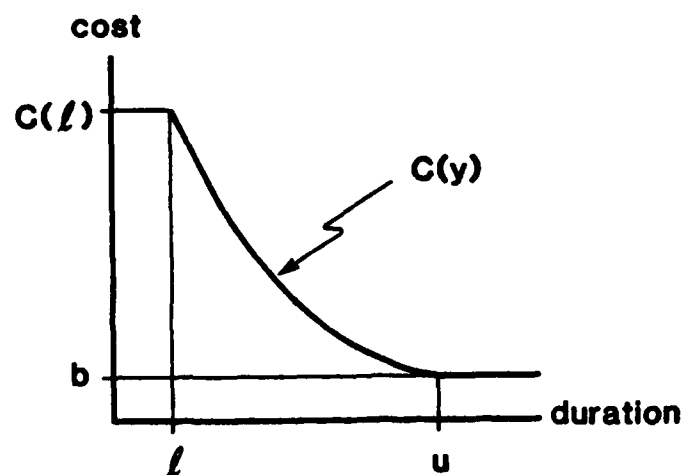


Figure 3. The Quadratic Function $C(y) = b + \beta(u-y)^2$; for $l \leq y \leq u$.

$0 = t_1 \leq t_{i_2} \leq t_{i_3} \leq \dots \leq t_n = T_s$, where t_{i_k} corresponds to the k th earliest node. Let $D(\tau)$ denote the sum of the derivatives C'_{ij} that are "in progress" at time τ . Then the given schedule of activities is optimal iff $D(\tau)$ is constant for all $\tau \in [0, T_s]$.

There are two remarks to be made about that result. First, it was proved by elementary variational-type arguments. Second, though it characterized the optimum, it gave no procedure for achieving it. The following development pertains to these two remarks.

Note that, since C_{ij} is quadratic decreasing in the interval $[l_{ij}, u_{ij}]$, then constant over $[u_{ij}, \infty]$, it is convex. Add to this that all the constraints are linear, and the conclusion immediately follows that the necessary conditions of Kuhn and Tucker^[16] for nonlinear programming are also sufficient. It is a simple matter to verify that these conditions translate directly into the condition $D(\tau) = \text{constant}$, $\tau \in [0, T_s]$; details of the proof may be found in [11]. This gives a more direct proof, albeit non-elementary.

As to the problem of algorithmic solution, ref. [11] also gives such an algorithm. It is based on the following results:

Proposition 1 The necessary and sufficient conditions for optimality stated above are equivalent to the conditions:

$$\begin{aligned} & -a \quad \text{for } i = 1 \\ \sum_j (d_{ji} - d_{ij}) = & \quad 0 \quad \text{for } i \neq 1, n \\ & a \quad \text{for } i = n \end{aligned} \tag{6}$$

where d_{ij} is the (normalized) reduction in activity (ij) , and a is some constant.

As a preliminary to the next result, let the "critical subnetwork (CSN)" denote the set of longest paths in the network. It is easy to establish that we may deal with the optimal "incremental" reductions $\{d_{ij}^{(r)}\}$ at the r th iteration, in place of the "total" reductions $\{d_{ij}\}$. The following two assertions ensure the solvability of the system of equations in the $\{d_{ij}^{(r)}\}$ unknowns.

Proposition 2 If the CSN contains K arcs, there shall be K simultaneous linear equations relating the values of the individual (incremental) reductions $\{d_{ij}^{(r)}\}$ at the r th iteration to the constant $a^{(r)}$.

The proof of this theorem rests on the fact that if the CSN has $m(\leq n)$ nodes and K arcs, there are $m-1$ independent equations (6) (the first equation is discarded), and exactly $K-m+1$ "fundamental loop" equations, for a total of K independent equations in the K unknowns. We shall refer to this system of linear equations notationally as

$$BD = a^{(r)} \underline{e}_{m-1} \quad (7)$$

where B is a $K \times K$ matrix of entries $0, \pm 1$; D is a $K \times 1$ column vector of $\{d_{ij}^{(r)}\}$, and the vector \underline{e}_{m-1} is a vector of zeros except in position $m-1$, where it has entry 1.

Proposition 3 The system (7), in the "incremental" reductions $\{d_{ij}^{(r)}\}$, possesses a unique solution.

We are now assured of the (unique) determination of $d_{ij}^{(r)}$ as a function of $a^{(r)}$, say $d_{ij}^{(r)} = v_{ij}^{(r)} a^{(r)}$. It remains to determine $a^{(r)}$. This

is accomplished by remarking that the current CSN can be "compressed" unabatedly until one of the following two eventualities occur:

- (i) Another (currently non-critical) path becomes "critical", or
- (ii) The specified duration T_s is reached.

In the first eventuality the CSN must be augmented by the new path (or paths); hence, the current system of equations (7) are no longer valid and must be updated. In the second eventuality, we terminate, since the optimum is in hand.

Denote the permissible reduction to eventuality (i) by $a_1^{(r)}$, and the permissible reduction to eventuality (ii) by $a_2^{(r)}$. Then, clearly,

$$a^{(r)} = \min \{a_1^{(r)}, a_2^{(r)}\}. \quad (8)$$

Back substitution into the expressions for $d_{ij}^{(r)}$ yields the respective values.

The suggested algorithm may be briefly sketched as follows.

Step (0) Set each activity at its upper bound $x_{ij} = u_{ij} \quad \forall (ij) \in A$.

Compute the node realization times $\{t^{(0)}\}$ and define the CSN.

Denote the set of arcs in the CSN by $K^{(r)}$ in iteration r .

Step (1) Compute $d_{ij}^{(r)}$ in terms of $a^{(r)}$; $(ij) \in K^{(r)}$, by solving the system of linear equations $B^{(r)} D^{(r)} = a^{(r)} \underline{1}_m$.

Step (2) Compute $a^{(r)} = \min(a_1^{(r)}, a_2^{(r)})$ if $K < A$, and $a^{(r)} = a_2^{(r)}$

if $K = A$ (i.e., if all activities in A are critical).

Step (3) Compute $T^{(r)} = T^{(r-1)} - a^{(r)}$ and return to Step (1) with $r = r + 1$ if $T^{(n)} > T_s$; otherwise halt.

Finally, we have

Proposition 4 The stated algorithm yields the desired solution in a finite number of steps.

The suggested procedure is "straightforward" except for two rather complex operations: first the inversion of the matrix $B^{(r)}$ of coefficients of (7), and, second, the determination of $a_1^{(r)}$. The first is an operation of order of complexity not exceeding $O(N^3)$; and the second is of $O(N^2)$. (Details of capitalizing in iteration r on the availability of the inverse of $B^{(r-1)}$ from the previous iteration, as well as the determination of the order of complexity of the operations involved are given in ref. [11].)

Resolving the problem for a specified completion time T_s also yields the approach to obtaining the complete optimal time-cost function for all feasible T_s (provided that Condition L is satisfied).

The procedure has been programmed in FORTRAN 4 on the NCSU's IBM 370/165. Details and documentation may be found in [18].

Case 2: Discontinuous Derivative

For more realism and practicality, we introduce two discontinuities in the derivative, one at ℓ and the other at u , as follows:

$$\begin{aligned} -dC/dy &= \begin{cases} C'(\ell) ; & \text{for } y = \ell^+ \\ \infty & ; & \text{for } y = \ell^- \end{cases} \\ -dC/dy &= \begin{cases} 0 & ; & \text{for } y = u^+ \\ C'(u) ; & \text{for } y = u^- \end{cases} \end{aligned}$$

The cost function and its derivative would then appear as shown in Figure 4. We shall refer to the derivative diagram as the kilter diagram (KD). In it we distinguish four regions:

- Region F (for "float"): $u < y < \infty; C' = 0$
 Region E : $y = u; C' < C'(u)$
 Region C (for "compressible"): $l < y \leq u; C'(u) \geq C' > C(l)$
 Region L (for "low bound"): $y = l; C' = \infty$

The algorithm specified below can be understood only in conjunction with the (necessary and sufficient) Kuhn-Tucker conditions for optimality, on the basis of which it is easy to construct the following "optimality table".

State of Activity Duration	Flow Condition	Domain of C_{ij}
$l_{ij} < y_{ij} = u_{ij} < t_j - t_i$	$f_{ij} = 0$	F
$l_{ij} < y_{ij} = u_{ij} = t_j - t_i$	$0 \leq f_{ij} < -C'_{ij}(u_{ij})$	E
$l_{ij} < y_{ij} = t_j - t_i \leq u_{ij}$	$C'_{ij}(l_{ij}) > f_{ij} = -C'_{ij}(y_{ij})$ $\geq -C'_{ij}(u_{ij})$	C
$l_{ij} = y_{ij} = t_j - t_i < u_{ij}$	$f_{ij} \geq -C'_{ij}(l_{ij})$	L

Table 1. Optimality Table

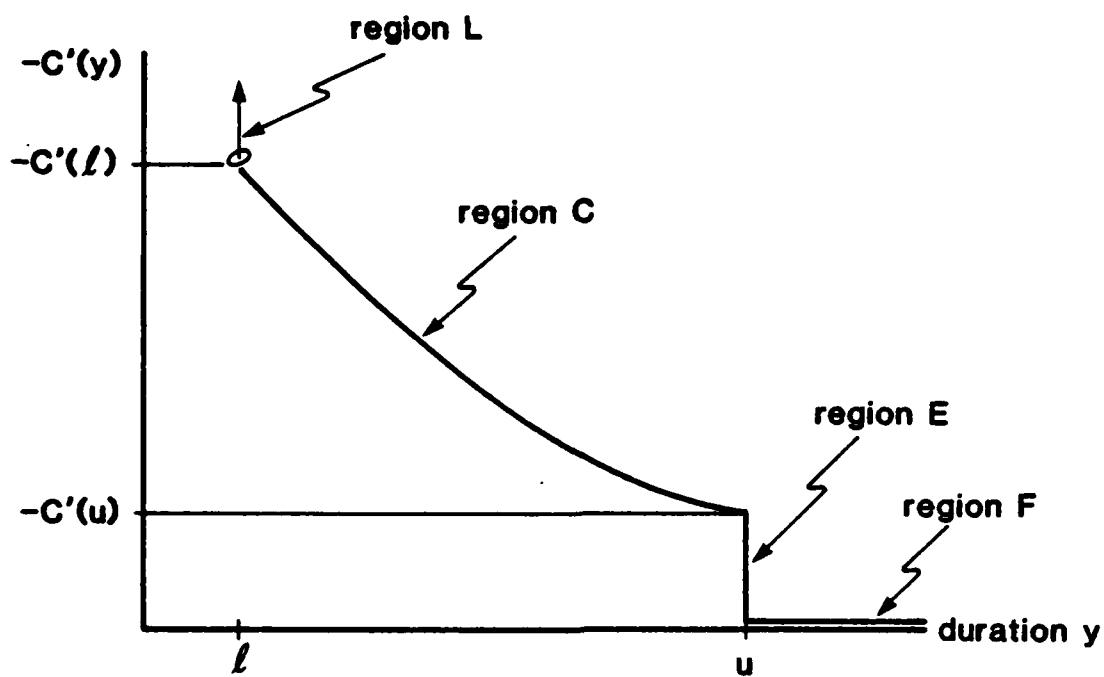


Figure 4. The KD for Case 2; $l \leq y \leq u$.

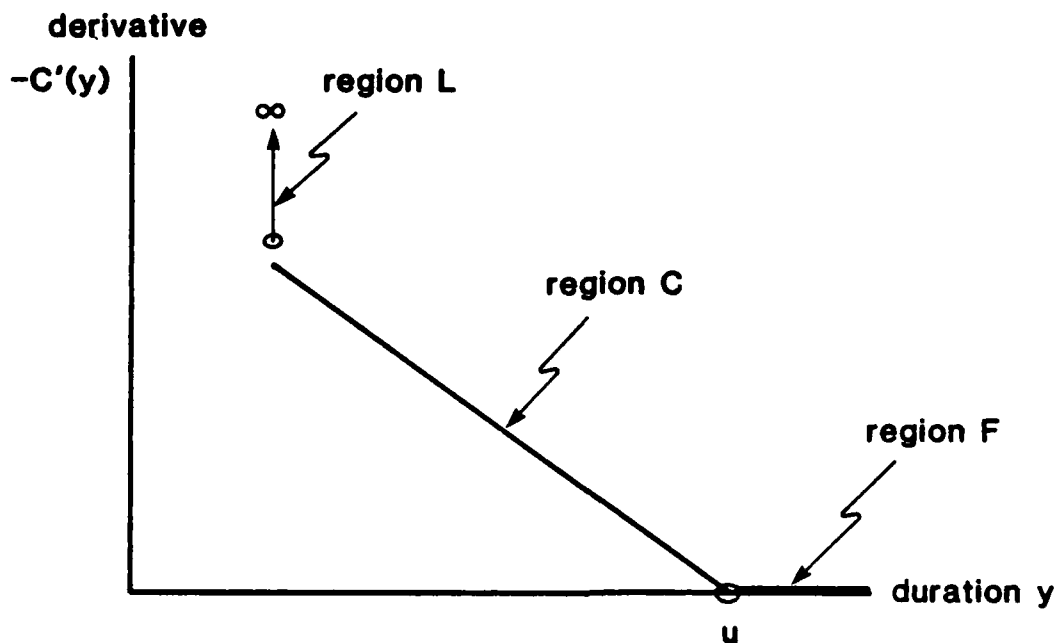


Figure 5. The KD for Case 1. (Region L is never reached, by Condition L.)

In passing, it is worth remarking that the applicability of the Kuhn-Tucker conditions to general convex (not necessarily quadratic) C_{ij} functions, as noted by Elmaghraby^[7], was first applied to quadratic cost functions by Kapur^[15] who achieved the KD and reasoned the optimality table shown above. Unfortunately, his subsequent development may not achieve the optimum, as demonstrated in [18]. This is mainly because of his strict adherence to the concept of "maximal flow cutset", which had validity in the case of linear cost functions (see [12]), but has no equivalent validity in the case of nonlinear functions, albeit it has some utility in parts of the calculations, as presently demonstrated.

Our algorithm rests on the following crucial observation: At iteration r , if all activities in the CSN were in State C then we would proceed in an identical fashion to our procedure of Case 1 discussed above. It is not difficult to see that Case 1 is, in some sense, a "degenerate" case of our present conditions, in which the KD has the shape of Figure 5, in which region E has disappeared and region L is never reached (because the duration l_{ij} is never realized, according to Condition L). But since now we must contend with regions E and L, the basic procedure of Case 1 should be modified to reflect the new concerns that have resulted from these two regions. In particular:

- (a) An activity in State E cannot be shortened, though the "flow" through it may be increased until it equals $C_{ij}(u_{ij})$, at which time the activity may be shortened.
- (b) An activity in State L is at its lowest possible duration, and therefore cannot be shortened at all, though "flow" through it may be increased indefinitely.

Careful study of the Optimality Table immediately reveals that any "flow" increase in the CSN is translatable (linearly) into change in duration of the individual activities in the CSN. Consequently, the limitations on $a^{(r)}$, the amount of permissible "compression" in iteration r , should be augmented by the two most recent considerations, as follows:

$a_3^{(r)}$: to be derived from the augmented flow such that activity (ij) moves from region E to region C (or equivalently, f_{ij} which was strictly $< C'_{ij}(u_{ij})$ is increased to $C'_{ij}(u_{ij})$).

$a_4^{(r)}$: to be derived from the augmented flow such that activity (ij) moves from region C to region L (or equivalently, f_{ij} which was strictly $> C'_{ij}(l_{ij})$ is decreased to $= C'_{ij}(l_{ij})$).

As before,

$$a^{(r)} = \min(a_1^{(r)}, a_2^{(r)}, a_3^{(r)}, a_4^{(r)})$$

in which $a_1^{(r)}$ and $a_2^{(r)}$ are determined as before.

We refrain from giving the full details of the algorithm; they may be found in [11]. However, we make the following remarks concerning its computability. We confine ourselves to iteration r , and therefore eliminate the explicit reference to the iteration number for the sake of simplicity.

A CSN may have arcs in states E, C and L. The bound a_3 on the permissible reduction in project duration, may be obtained by interpreting the arcs in state E as possessing capacities equal to $[C'_{ij}(u_{ij}) - f_{ij}] > 0$; $(ij) \in E$, and performing a standard flow augmentation step. Conceptually,

this brings the marginal value of "cohort activities" (i.e., activities that lie on a minimal capacity cutset containing (ij)) to the same level as that of activity $(ij) \in E$, and there would be no inequity in the valuation of the various activities.

The change in duration of activities in state C requires the solution of a system of K linear equations in K unknowns, similar to Case 1, Equations (7). Unfortunately, the matrix B no longer possesses the desirable property of ± 1 or 0 entries: its top $m-1$ rows will indeed have such entries, but the bottom $(K-m+1)$ rows (corresponding to the "fundamental loops") will have fractional entries. This is due to the non-zero slope of C_{ij} in region C, which relates the "flow" variables f_{ij} to the amount of reduction d_{ij} .

Finally, it must be noted that while each path in the CSN is shortened at each iteration, any individual activity need not follow such monotone behavior: an activity may be lengthened after having been shortened, though no activity will possess positive float if it had once been in the CSN. This is not a new result, since it is observed even in the linear cost case^[12].

The procedure has been programmed in FORTRAN 4 on the University's IBM 370/165. Details and documentation may be found in [18].

2. OPTIMAL LINEAR APPROXIMATIONS

It is a truism that project compression under linear cost functions of the form $C(y) = b - ay$; $l \leq y \leq u$; $a, b > 0$, is considerably easier to resolve optimally than under nonlinear cost functions. The above analysis, carried under the simplifying assumption of quadratic cost functions should amply demonstrate the fact, if such demonstration were needed! The natural question then is: what if $C(y)$ is not quadratic, though still convex decreasing as y increases from l to u . Can the problem still be analyzed? For instance, suppose

$$C(y) = a/(b+ky); \quad 0 \leq l \leq y \leq u < \infty; \quad a, b, k > 0. \quad (9)$$

What can be said about the optimum in this case?

One may wish to persist in applying the theoretical constructs of the exact solutions, which are indeed applicable in toto. Unfortunately, the KD will now possess a nonlinear segment in region C, which would necessitate the solution of a system of nonlinear equations in the "flows" $\{f_{ij}\}$ at each iteration; an onerous task at best.

The other alternative is to approximate the cost function $C(y)$ by a piecewise linear and convex function (i.e., linear spline) that is optimal in some sense. There are two immediate questions that present themselves: the first is to define the sense of the approximation itself, and the second is to define the criterion of optimality of the approximation. We elaborate on these two questions.

The approximation we seek should guarantee a deviation from the optimum value that does not exceed a prescribed proportional error δ . To see what this implies, let $C(Y) = \sum_{(ij) \in A} C_{ij}(y_{ij})$, and assume the optimum is achieved at the vector of durations $y^* = \{y_{ij}^*\}$, with corresponding value $C(Y^*)$. Of course, y^* is unknown, and we wish to approximate the value $C(Y^*)$. Let $H_{ij}(y_{ij})$ denote the piece-wise linear and convex approximation to $C_{ij}(y_{ij})$, and let $H(Y) = \sum_{(ij) \in A} H_{ij}(y_{ij})$ be the criterion function of the linear program (LP) defined by the constraints (2)-(4). The solution of this LP, which is achieved relatively easily, shall yield a vector of activity durations, which we denote by $\eta^* = \{\eta_{ij}^*\}$, and the corresponding value $H(\eta^*)$. Now, the requirement we impose may be stated as follows: Select the approximation H_{ij} to satisfy the inequality

$$|C(Y^*) - H(\eta^*)| \leq \delta |C(Y^*)| \quad (10)$$

for any prescribed value $\delta > 0$. Typically, δ is less than 1, chosen from the interval $[.01, .10]$. Restriction (10) simply ensures that the approximate optimum $H(\eta^*)$ shall not deviate from the true optimum $C(Y^*)$ by more than a small fraction of the value of the true optimum.

We now address the issue of the "measure of closeness" of the individual approximating (linear) functions $H(y)$ to the original $C(y)$: We adopt the "maximum norm" (Chebychev) criterion. In other words, we seek a piece-wise linear and convex approximating spline whose maximal deviation from the original function $C_{ij}(y_{ij})$ is minimal (i.e., optimal in the sense of Chebychev).

Our procedure comprises two basic steps: The first accepts the data of the original problem and the specified $\delta > 0$ and yields a value $\epsilon > 0$ that represents the bound on the maximal deviation between C_{ij} and H_{ij} . The second accepts ϵ and constructs for each function $C_{ij}(y)$ the approximating linear spline $H_{ij}(y)$ that deviates from $C_{ij}(y)$ by at most ϵ throughout the range of y .

The details of the construction are given in [9]. The procedure is programmed for the function $C_{ij}(y) = a_{ij}/(b_{ij} + k_{ij}y)$ for illustrative purposes.

II. ESTIMATIONS IN THE PERT MODEL

One of the main advantages of using network analysis for project planning and control is the ability to identify the activities that are critical to the achievement of the project objectives. In DANs, it is relatively easy to respond to questions such as: What is the critical path(s)? What are the most M critical activities? and so forth.

We seek to develop the analogous results in PANs, such as the PERT model. Clearly, one must phrase the questions in probabilistic terms such as: What path (or paths) is the most probable to be critical? Which activity (or activities) has the highest probability of being critical? What is the probability that a particular path is critical? Which (minimal) paths have a total probability of being critical at least β , etc.?

In the following sections we formalize these intuitive notions and develop procedures for approximating their measures. We concentrate in this brief report on delineating the fundamental concepts adopted in the approximations used, leaving the detailed accounts to a companion report in preparation.

Some Definitions and Basic Concepts

Let P be the set of paths in the AN and let p_h denote the h th path (from node $\underline{1}$ to node \underline{n}), and let $Z(p_h)$ denote its duration, $h = 1, 2, \dots, p$. The criticality index (CI) of a path p_h is denoted by $CR(p_h)$, and defined by

$$CR(p_h) = \Pr(Z(p_h) \geq Z(p_q) \text{ for all } p_q \in P) \quad (11)$$

where $\text{Pr}(\cdot)$ means the "probability of", and the duration

$$Z(p_h) = \sum_{(ij) \in p_h} Y_{ij} \quad (12)$$

Here, Y_{ij} represents the duration of activity (ij) , a random variable (r.v.) presumed of known probability distribution function (pdf).

$\text{CR}(p_h)$ has been estimated by several analytical and Monte Carlo sampling techniques, the latter ranging from the "crude" to the very "sophisticated" ([1, 19, 20]). Our objective is to present analytical approaches to the determination of $\text{CR}(p_h)$ and the other measures stated above.

The criticality of an activity (ij) , $1 \leq i < j \leq n$, is defined as the sum of the CIs of the paths containing it. We denote the CI of activity (ij) by the symbol $\text{CA}(ij)$; hence,

$$\text{CA}(ij) = \sum_{p_h} \text{CR}(p_h) ; \quad (ij) \in p_h \quad (13)$$

The procedure described in the following section relies heavily on the iterative algorithm of Dodin [4] for the estimation of the pdf of the project completion time, which is summarized next because of its relevance to computing.

1. THE APPROXIMATION OF THE PDF OF PROJECT COMPLETION TIME

Dodin [4] developed a system of computer programs that accomplish four tasks:

1. Generate random AN of a pre-specified number of nodes N and arcs A ; i.e., a network from the set of networks with this n and $|A|$, in which all are equally probable. The procedure represents a slight improvement over that of Herroelen and Caestecki^[13]. Basically, there are two approaches, which are best discussed relative to the $N \times N$ adjacency matrix. The first is the "deletion method", which starts with the upper triangle above the main diagonal full, then sequentially eliminates entries of 1 randomly, subject to the restriction that every node is connected to both the origin as well as the terminal nodes until the desired number is left. The second is the "addition method", which reverses the view and starts with an empty upper triangle that is to be filled sequentially until the desired number of arcs are present, subject to the same constraint.

The choice of the method to implement depends, obviously, on the density of the network. The deletion method is preferred if $|A| \geq n(n-1)/4 + 1$.

2. Discretize any given continuous pdf. Three approaches have been tried; and the most efficient in terms of accuracy and computer time is a hybrid of the last two. Let m denote the number of discrete points that represent $F(\cdot)$. (i) the first method assumes that both the location x_k and the probability mass $p(x_k) \equiv p_k$ of occurrence of x_k are unknown, $k = 1, \dots, m$. It is desired to determine these $2m$ unknowns by equating the first $2m$ moments of the discrete approximation to the (given) theoretical df:

$$\sum_{k=1}^m x_k^r p_k = \mu_r, \quad \text{for } r = 0, 1, \dots, 2m-1 \quad (14)$$

where

$$\mu_r = E(x^r) = \int_0^{\infty} y^r dF(y) ; \quad \text{the } r\text{th moment .}$$

Equations (14) may be represented in matrix form as

$$VP = E$$

where V is the well-known Vandermonde matrix of dimensions $2m \times m$, P is the probability vector with m components, and E is the vector of $2m$ moments $\{\mu_n\}$. Two methods were tried to solve this system of nonlinear equations, but, unfortunately, neither succeeded for $m > 8$. This approach was then abandoned. (ii) The second method may be termed "the equal interval method", in which the (finite) range of the arc duration is divided into m equal intervals of width Δ each. The finiteness of the range is secured by defining two points ℓ and u by

$$\Pr(X < \ell) = 0.0005 = P(x > u) .$$

With the intervals defined, it is easy to determine the probabilities $\{p_k\}$, assumed to be associated with $\{x_k\}$ at the min-points of their respective intervals. This method proved quite satisfactory for df's that possess no sharp peaks or severe skewness. The method is also convenient for the use of the Fast Fourier Transform (FFT) method in the successive approximation discussed below.

(iii) The third method divides the (finite) range into intervals of unequal length but of equal probabilities ($= 1/m$). It is a simple matter to determine the intervals through the inverse function $F^{-1}(\cdot)$, and proceed assuming the probability concentrated as the mid-point of each interval.

A hybrid of approaches (ii) and (iii) would use equal intervals where the df is flat, or nearly so, and reverts to equal probabilities where it peaks.

3. Reduce the AN to its irreducible form, through the operation of addition (\equiv convolution operation) of arcs in series and multiplication (\equiv maximum operation) of arcs in parallel. We wish to make two remarks on this step. The first is that the number of discrete points m is usually held fixed beyond a certain point, and a reduction operation that yields a number of points larger than m must be "folded back" to only m points. This introduces the first source of error in the approximating procedure. The second remark is that it is in this step that the equal interval method helps because of our ability to use the FFT in the reduction process.

4. Approximate the irreducible network. This is the "heart" (central element) of the computer package and embodies the concept of independence that is usually invoked relative to paths into any node $i \in N$.

The iterative progression over the nodes of the network employs the convolution of $F_i(t)$ with $F_{ij}(t)$, where $F_i(t)$ denotes the pdf of T_i , the time of realization of node $i \in B_j$, and $F_{ij}(t)$ is the pdf of X_{ij} , the duration of activity (ij). The convolution operation can be performed by either the usual formula or through the use of FFT. Both approaches were tested and preference is given to the usual formula.

The accuracy of the approximation was tested relative to two measures:

(i) The max-absolute deviation $\max_t |F_n(t) - \hat{F}_n(t)|$

(ii) The average value of the absolute deviation

$$\int_0^{\infty} |F_n(t) - \hat{F}_n(t)| dt .$$

In these expressions, the "exact" pdf $F_n(t)$ was secured through MCS.[†] The results are most encouraging: (a) For networks up to 60 nodes and 150 activities, the max. deviation was less than .08 and the average deviation less than .03. (b) The max. value of the absolute deviation occurs within the low 30% of the range of the r.v. This is reassuring, since in the study of ANs the realizations of greatest interest are those in the "right tail" of the distribution; i.e., in the high 30% of the range. (c) The sampled distribution $F_n(t)$ (via MCS) converges toward the approximate distribution $\hat{F}_n(t)$ as the sample size increases! This is an unexpected result, since it implies that the sampled df is the "inexact" one! (d) The processing time of the approximate procedure are quite reasonable, being of the order of 30 secs. for networks of size $(N,A) \leq (60,120)$ on the AMDAHL V-7.

2. THE ACTIVITY CRITICALITY INDEX

Let n_j denote the in-degree of node j , and assume the arcs into j to be numbered in increasing order, the same as their originating nodes; see Figure 6.

[†]Monte Carlo Sampling.

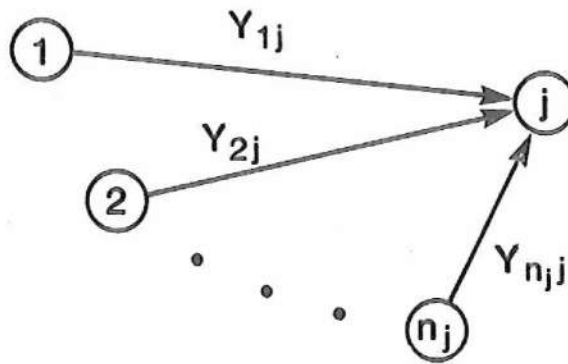


Figure 6. The scheme of numbering arcs into \underline{j} .

We define the cutset C_j at node \underline{j} to be the set of arcs resulting from the partitioning of the set of nodes N into two mutually exclusive subsets S_j and T_j where

$$S_j = \{i \in N : i < j\} \quad \text{and} \quad T_j = \{i \in N : i \geq j\}$$

Hence

$$C_j = \{(hk) \in A : h \in S_j \text{ and } k \in T_j\} \quad (15)$$

Let T_i^f denote the duration of the longest path forward from node $\underline{1}$ to node \underline{i} , and T_j^b denote the duration of the longest path backwards from node \underline{n} to node \underline{j} . Clearly, both T_j^f and T_j^b are r.v.'s. Let W_{ij} denote the duration of the longest path containing arc (ij) ; then

$$W_{ij} = T_i^f + Y_{ij} + T_j^b \quad (16)$$

$$\text{where } T_1^f = \max_{p_{1i}} \{Z(p_{1i})\} \text{ and } T_j^b = \max_{p_{Nj}} \{Z(p_{Nj})\} \quad (17)$$

and p_{ij} denotes a path from i to j .

$$\text{Proposition 5 } W_{ij} = \max_{\substack{p_h \\ (ij) \in p_h}} \{Z(p_h)\} ; Z(p_h) \text{ as defined}$$

in (12).

The proof is by direct substitution in the definition of W_{ij} in (16).

Now, the exact value of the CI of arc (ij) is given by (see (13) and (11)):

$$\begin{aligned} CA(ij) &= \sum_{(ij) \in p_h} CR(p_h) \\ &= \sum_{(ij) \in p_h} \Pr(Z(p_h) \geq Z(p_q) \text{ for all } p_q \in P) \end{aligned} \quad (18)$$

Let $L(ij)$ denote the subset of paths that contain arc (ij) . Then, clearly, $CA(ij)$ measures the "weight" attached to the event that any $p_h \in L(ij)$ is longer than any other path in the network. Unfortunately, it is extremely difficult to calculate $CA(ij)$ directly from (18) because of the need to identify the set of all paths in the network and to calculate the corresponding CR's. To obviate that need, we appeal to the concept of "cutset" defined above. Let

$$CAP(ij) = \Pr(W_{ij} \geq W_{kl} \text{ for all } (kl) \in C_j) \quad (19)$$

where C_j is as defined in (15). Close scrutiny reveals that $CAP(ij)$ measures the probability that the maximum of the paths in the subset $L(ij)$ is longer than the maximum of the paths not containing (ij) , (i.e., in the complementary set $\bar{L}(ij) = P - L(ij)$). These latter paths are precisely the paths that contain all the arcs in C_j except arc (ij) . It is demonstrated in [6] that $CAP(ij)$ always underestimates $CA(ij)$. But for the moment, we ascertain the iterative manner in which C_j is obtained from C_{j+1} .

$$\text{Proposition 6} \quad C_j = C_{j+1} + \{(ij) \in A : i < j\} - \{(jk) \in A : k > j\}$$

with the initial condition

$$C_N = \{(iN) \in A\}$$

where A is the set of activities in the network.

The proof is achieved by induction, starting with node $N-1$.

$$\text{Proposition 7} \quad CAP(ij) \leq CA(ij) \quad \text{for all } (ij) \in A$$

We argue heuristically as follows. If any path $p_h \in L(ij)$ is longer than any path $p_q \in \bar{L}(ij)$, then a fortiori, $\max\{Z(p_k) : p_k \in L(ij)\}$ is longer than $\max\{Z(p_q) : p_q \in \bar{L}(ij)\}$. Therefore, the set on which $CA(ij)$ is defined contains the set on which $CAP(ij)$ is defined. Moreover, the sum of probabilities defining $CA(ij)$ is no less than the probability of the union of the events in the set, which is no smaller a set than that defining the probability of $CAP(ij)$. Consequently, $CAP(ij) \leq CA(ij)$.

Proposition 8 For any node $i = 1, n$,

$$\sum_{j \in B(i)} CA(ji) = \sum_{j \in A(i)} CA(ij) \quad (20)$$

where the sets $A(i)$ and $B(i)$ denote the sets of nodes connected to i and occurring after it and before it, respectively.

The proof is accomplished by defining the CI of node i , $CN(i)$ as follows:

$$CN(i) = \sum_{\substack{p_h \\ i \in p_h}} CR(p_h) .$$

That is, the CI of a node is the sum of the CIs of the paths containing that node. Then it is easy to show, by appealing to definitions, that $CN(i)$ is equal to each side of equality (20).

Two immediate consequences of Proposition 8 follows:

$$\text{Corollary 1} \quad CN(1) = CN(n) = \sum_{p_h \in P} CR(p_h) \geq 1.0 \quad (21)$$

The two equalities in (21) are rather obvious; and the last inequality follows from the definition of $CR(p_h)$ as probability, and the fact that the paths in the network are not necessarily independent. (Equality to 1.0 is achieved only when the paths are independent and, in the case of discrete pdf's, no two paths are critical simultaneously.)

Corollary 2 The CI of any path is equal to the CI of any unique arc on the path (i.e., an arc that belongs to no other path). Moreover, all unique arcs on the same path have the same CI.

We now concern ourselves with the computation of $CAP(ij)$. Let $V(ij)$ denote the maximum duration of paths not containing arc (ij) .

$$V(ij) = \max_{\substack{(hk) \in C_j \\ (hk) \in \bar{L}(ij)}} \{W_{hk}\}$$

Consequently, from (19), we may write

$$CAP(ij) = \Pr[W_{ij} \geq V(ij)]$$

in which each W_{rs} is given by (16), $(rs) \in C_j$. Thus the problem of determining the value of $CAP(ij)$ reduces to calculating the pdf of the r.v.'s T_r^f and T_s^b , $(rs) \in C_j$, and performing the necessary convolution and maximum operations. But the calculation of the (approximate) pdf's of T_r^f and T_s^b is precisely the problem discussed in [4]. As is mentioned there, the causes for the errors in estimation of these pdf's are three:

(i) the discretizing of continuous distributions (if any); (ii) the assumption of independence of paths; and (iii) the reduction of the domain of the computed pdf's to a predetermined (small) number of discrete points. Most importantly, in [4] it is demonstrated that the approximation to the pdf of W_{rs} may either overestimate or underestimate the true pdf. Consequently, the approximation to $CAP(ij)$, denoted by $ACAP(ij)$, cannot be asserted to be an underestimate of $CA(ij)$, though empirical evidence in [4] indicates that the approximation is excellent.

Finally, it was demonstrated in Cor. 1 that $CN(1) \geq 1.0$. In case $CN(N) > 1$, which is almost always true, it is advantageous to normalize all CIs, for arcs and nodes, by dividing throughout by $CN(N)$, because such normalization reduces the maximum error in the estimation

of activity CIs. Indeed, the original maximum error =

$\max_{(ij) \in A} (|CA(ij) - CAP(ij)|)$, while the normalized maximum error is

precisely $1/CN(N)$ of its value. A minor benefit of such normalization is that the various CIs may be thought of now as probabilities, which was not possible before.

3. THE HIGHEST K-CRITICAL PATHS

The title implies either of the following two problems:

(i) the identification of the minimum set of paths whose probability of being "critical", i.e., that any member of the set is of no shorter duration than any path in the network that is not in the set, is at least β ; $0 < \beta < 1$, (typically $\beta \geq 0.50$); (ii) the identification of the K "most critical" paths in their rank order, i.e., the path(s) with the highest probability of being critical, the path(s) with the next highest probability of being critical, and so forth to the K th ranking path(s).

The theoretical discussion presented below leads to an approximating procedure that emulates that used in DANs to identify the first K CPs.

Let $P(j) \subseteq P$ denote the set of paths ending in node j ; $p_j^k \in P(j)$ denote the k th ranked "CP" ending in node j , and $Z(p_j^k)$ its duration. We say that p_j^k dominates p_j^r in probability, denoted by $p_j^k \geq p_j^r$ if

$$\Pr[Z(p_j^k) \geq Z(p_j^r)] \geq \Pr[Z(p_j^r) \geq Z(p_j^k)] \quad (22)$$

The following assertion is an immediate consequence of the above definition.

Proposition 9 Let p_i^1 and p_i^2 be two paths ending in node i such that $p_i^1 \succeq p_i^2$. Then $p_j^1 = [p_i^1 o(ij)] \succeq [p_i^2 o(ij)] = p_j^2$, where "o" denotes the catenation (or extension) of the path from the right.

In words, this Proposition asserts that adding activity (ij) to both paths does not alter their relative ranking, which is eminently plausible.

Unfortunately, dominance in probability as defined in (22) is, in general, intransitive, contrary to prima facie expectations.

That is, if $p_i^1 \succeq p_i^2$ and $p_i^2 \succeq p_i^3$, then the relation $p_i^1 \succeq p_i^3$ need not be true in general, but is true for symmetric distributions. To see this, note that the dominance relation $p_i^1 \succeq p_i^2$ implies that $\Pr[Z(p_i^1) \geq Z(p_i^2)] \geq 1/2$. We thus have

$$\Pr[Z(p_i^1) - Z(p_i^2) - (\mu_1 - \mu_2) \geq -(\mu_1 - \mu_2)] \geq 1/2 \quad (23)$$

where $\mu_r = E[Z(p_i^r)]$. Now, assuming symmetric pdf's, it is clear that the r.v. $[Z(p_i^1) - Z(p_i^2)]$ is also symmetric about its mean $(\mu_1 - \mu_2)$, and inequality (23) implies that $-(\mu_1 - \mu_2) \leq 0$; i.e., $\mu_1 \geq \mu_2$. Similarly, the dominance relation $p_i^2 \succeq p_i^3$ implies that $\mu_2 \geq \mu_3$ under the assumption of symmetric pdf's. We therefore conclude that $\mu_1 \geq \mu_3$. Now reversing the argument we conclude that $\Pr[Z(p_i^1) \geq Z(p_i^3)] \geq 1/2$, which finally implies that $p_i^1 \succeq p_i^3$. We have just proved

Proposition 10 If the paths to a node possess symmetric pdf's, then dominance in probability as defined in (23) is transitive. Furthermore, $p_i^1 \geq p_i^2 \implies \mu_1 \geq \mu_2$, where $\mu_r = E[Z(p_i^r)]$.

The utility of these conclusions is evident for the result sought. High-numbered nodes are the ones most prone to having a large number of paths (from node 1) leading into them. (One can easily verify that node 2 has 1 path, node 3 has ≤ 2 paths, node 4 has ≤ 4 paths, node 5 has ≤ 8 paths and node 6 has ≤ 16 paths. In general, node i has $\leq 2^{i-2}$ paths leading into it from node 1.) But these paths are precisely the ones whose pdf's may be approximated by symmetric distributions (usually the normal pdf). Consequently, they are the ones to which the assumption of transitivity of dominance is appropriate. Lower-numbered nodes are not in need for such approximation since their paths may be explicitly enumerated and ranked.

The algorithm alluded to at the beginning of this section is now apparent, and its gist is as follows. For any node j, let its immediately preceding nodes be i_1, i_2, \dots, i_s ; and suppose that the first K CPs to node i_r have been identified as $p_{i_r}^1 \geq p_{i_r}^2 \geq \dots \geq p_{i_r}^k$. To determine p_j^1 , the most critical path to node j, we need to compare only the s topmost r.v.'s $\{Z[p_{i_r}^1 o(i_r, j)]; r = 1, 2, \dots, s\}$ and rank them. The highest ranking path (in probability) is p_j^1 . Now, the second ranking path in this set is compared with the second ranking path in the set of paths to which the topmost path belonged; and the higher ranking (in probability) between these two paths is p_j^2 ; and so on for

p_j^k , $k = 3, 4, \dots, k$. At termination, the answer to the two problems posed at the beginning is in hand:

- (i) The K most CPs of the network are precisely $p_n^1, p_n^2, \dots, p_n^K$.
- (ii) The minimum set of paths whose probability is no less than β is easily obtained from the most critical paths into node n .

Computing experience with this approximating procedure, and comparison with the results obtained from extensive MCS in determining the most critical three paths in networks of varying sizes reveal three significant facts: (i) the match between the approximating procedure and MCS decreases with path rank: there was 93% matches in path rank 1, 71% matches in rank 2, and only 57% matches in rank 3; (ii) the approximate procedure consumes significantly less time than MCS (approximately an order of magnitude less); (iii) In several instances, the set of paths identified by both the approximate procedure and MCS were identical, but the rank of the paths within the two sets was different. This is encouraging since it implies that both approaches would identify the same set of arcs as critical; (iv) Experimentation with MCS was necessarily aborted at small-size networks ($n = 30$ and $|A| = 90$), since larger networks would have required inordinately large amounts of time. This phenomenon was not experienced by the approximate procedure because of the difference in complexity between the two: MCS requires the enumeration, identification, and comparison of paths, which is a process that is of exponential complexity, while the approximate procedure is of complexity $O(n^2)$.

III. REDUCIBILITY OF ANs

The third area of investigation is related to the problem of "reduction" of ANs, which rears its head in more than one investigation in the context of ANs. (For a description of three such investigations, see Colby and Elmaghraby^[3].) Here we limit our attention to the PERT model and ask the question: What is the d.f. of the time of realization of the "terminal node" of the network (which signifies the completion time of the project)? Now, it is well-known that two activities in series may be collapsed into one activity whose d.f. is given by the convolution of the two individual d.f.'s. On the other hand, two activities in parallel may be collapsed into a single activity whose d.f. is given by the product of the two individual d.f.'s. If the original network can be collapsed into a single activity (1,n) then, indeed, the analytical form of the d.f. of the duration of the project is in hand. Unfortunately, the irreducibility of such PERT networks prohibits such (conceptually easy) analytical determination, which, in turn, gave rise to various approximating or bounding procedures discussed in II above.

Consequently, we say that a digraph is reducible if either of the following two conditions is satisfied:

- (a) There exists at least one path with node(s) of in-degree one and out-degree one (i.e., the path contains two arcs in series)
- (b) There exist at least two paths "in parallel"; i.e., there are two distinct nodes i and j , $1 \leq i < j \leq n$, and two distinct paths from i to j with the property that if there is an

intermediate node on either path between i and j , then it is of in-degree and out-degree one.

The "reduction process" amounts to the collapsing of two arcs into one, starting with arcs in series (the process may alternate between combining arcs in series, then arcs in parallel, then arcs in series that have been created by the arcs in parallel; etc.). A digraph is said to be completely reducible if it is collapsible to a single arc joining nodes 1 and n . Otherwise, we terminate with a graph that is irreducible (which is shorthand for "not completely reducible"). Then, evidently, both conditions (a) and (b) are not satisfied.

The problem of irreducibility of ANs has been recognized by every researcher in the field since the classical paper of Malcolm et al^[17] on the PERT model; (for citations, see Elmaghraby's book [8], Ch. 4). Consequently, it is natural to inquire into the conditions under which a day is irreducible. To this end we introduce some definitions and notation.

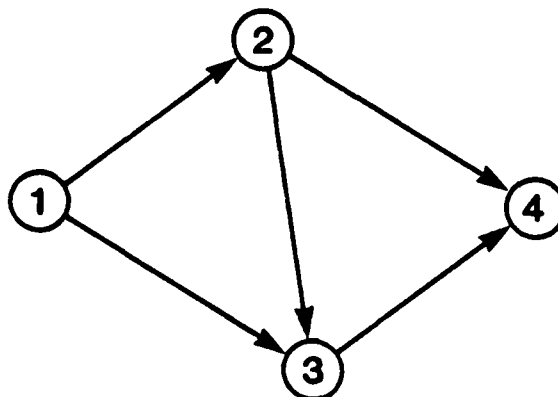


Figure 7. The interdictive graph (IG).

The "Interdictive Graph" (IG) is the graph shown in Figure 7.

Evidently, it is irreducible

We write $IN(\underline{i})$ and $OUT(\underline{i})$ as shorthand for the in-degree and the out-degree of node \underline{i} , respectively.

We write $NS(\underline{a})$ and $NE(\underline{a})$ for the "start node" and "end node" of arc $\underline{a} \in A$, respectively.

By a descendent of node \underline{i} we mean a node $j > \underline{i}$ which is connected from \underline{i} by an arc or a path.

The set of all nodes that connect to node \underline{j} by a path is denoted by $P(\underline{j})$; i.e., $P(\underline{j}) \triangleq \{i \in N : \underline{i} < \underline{j} \text{ and } \underline{i} \text{ connects to } \underline{j} \text{ by a path}\}$.

Properties of Irreducible Digraphs (IDG)

The following properties of IDG's are easy to verify. For the sake of brevity we shall not clutter this note with their proofs. They are numbered consecutively from the previous two properties:

- (3) The number of nodes $|N| \geq 4$.
- (4) Either $OUT(\underline{1}) = 1$, in which case $IN(\underline{2}) = OUT(\underline{1}) = 1$ and $OUT(\underline{2}) \geq 2$; or $OUT(\underline{1}) \geq 2$. Therefore, without loss of generality, we can take the IDG to start at node $\underline{1}$ whose $OUT(\underline{1}) \geq 2$.
- (5) Either $IN(\underline{n}) = 1$, in which case $OUT(\underline{n-1}) = IN(\underline{n}) = 1$ and $IN(\underline{n-1}) \geq 2$; or $IN(\underline{n}) \geq 2$. Similarly, we can assume that $IN(\underline{n}) \geq 2$.
- (6) There exists a smallest-numbered node $\underline{i}_2 \neq \underline{1}, \underline{2}$ such that $\underline{2}, \underline{j} \in P(\underline{i}_2)$ and the paths $\Pi_1^{(2)} = (1, 2, \dots, \underline{i}_2)$ and $\Pi_1^{(3)} = (1, j, \dots, \underline{i}_2)$ are independent (i.e., they have no intermediate node in common). (Note that the existence of such a node is guaranteed by the fact that both $\underline{2}$ and \underline{j} are $\in P(\underline{n})$; see Property 2.)
- (7) For all $\underline{i} \neq \underline{1}, \underline{n}$, $IN(\underline{i}) + OUT(\underline{i}) \geq 3$; hence there are no arcs in (simple) series.

The main results of Elmaghraby and Dodin [10] is the following:

Proposition: A digraph is irreducible iff it contains the IG.

The proof of this assertion is elementary and rests on the properties of the IG enumerated above.

Two questions flow naturally from this result. The first is: How to detect (efficiently) the presence of the IG, and if there is more than one, determine their count and their identity? And the second is: What is the most economical way (in the sense of minimum arcs) to "fix" in order to render the network completely reducible.

The issue of detection is easily answered by: do the reduction (which is easily defined in polynomial time ($O(m)$), and if the trivial network (of only one l, n arc) is not achieved, then the IG must exist. Henceforth, we refer only to the remaining two questions.

It is our contention that either of these two questions poses a problem that is NP-Hard. Colby^[2] demonstrated that both problems are in the class NP. He also proposed a heuristic procedure that is of polynomial complexity ($O(n^4)$) that dominates an earlier procedure by Dodin^[5]; see the paper by Colby and Elmaghraby^[3] for details and examples.

One final remark. Despite the fact that interest in the minimal number of arcs to "fix" springs from the desire to secure the exact (or approximate) pdf of the time of realization of node j , it is easy to show (as was demonstrated by Dodin^[5]) that fixing the minimum number of arcs is also useful in bounding the pdf from below. That is, if one is not interested in determining the exact pdf (through multiple integration

over the "fixed" arcs), but rather is interested in deriving l.b. of the pdf, then the identification of the minimum number of fixed arcs help generate tighter bounds.

REFERENCES

1. Burt, J.M., Jr. and Gorman, M.B., "Conditional Monte Carlo: A Simulation Technique for Stochastic Network Analysis", Management Sci. 18(3), 207-217.
2. Colby, A.H., (1984), "On the Complete Reduction of Acyclic Networks", Unpublished Master's Thesis, N.C. State University, Raleigh, N.C.
3. Colby, A.H. and S.E. Elmaghraby, (1984), "On the Complete Reduction of Directed Acyclic Graphs", OR Report No. 197, Graduate Program in Operations Research, N.C. State University, Raleigh, N.C.
4. Dodin, B.M., (1980), "On Estimating the Probability Distribution Function in PERT-Type Networks", OR Report No. 153 (Revised), Graduate Program in Operations Research, N.C. State University, Raleigh, N.C.
5. Dodin, B.M., (1983), "Bounding the Project Completion Time Distribution in PERT Networks, Research Report, School of Business Administration, University of Wisconsin-Milwaukee.
6. Dodin, B.M. and Elmaghraby, S.E., "Approximating Criticality Indices in PERT Networks", OR Report No. 177, April 1981, NCSU at Raleigh, Accepted for publication, Management Sci., 1984.
7. Elmaghraby, S.E., (1968), "The Determination of Optimal Activity Duration in Project Scheduling", J. Ind. Eng., 19, 48-51.
8. Elmaghraby, S.E., (1977), Activity Networks: Project Planning and Control by Network Models, John Wiley & Sons, Inc.
9. Elmaghraby, S.E., (1980), "A Note on Optimal Chebychev Approximations in Mathematical Programming", OR Report No. 156, Graduate Program in Operations Research, N.C. State University, Raleigh, N.C.
10. Elmaghraby, S.E. and Dodin, B.M., (1979), "Irreducibility of Acyclic Digraphs", OR Report No. 154, Graduate Program in Operations Research, N.C. State University, Raleigh, N.C.
11. Elmaghraby, S.E. and Salem, A.M., (1982), "Optimal Project Compression Under Quadratic Cost Functions", in Applications of Management Sci., Vol. 2, 1-39.
12. Fulkerson, D.R., (1961), "A Network Flow Computation for Project Cost Curve", Management Sci., 7, 167-178.
13. Herroelen, W. and Caestecke, G., "The Generation of Random Activity Networks", March 1979, Report No. 7906, Department of Applied Economic-Sciences, Katholieke Universiteit, Leuven, Belgium.

14. Jewell, W., (1965), "Risk Taking in Critical Path Analysis", Management Sci., 11(3), 438-443.
15. Kapur, K.C., (1973), "An Algorithm for Project Cost-Duration Analysis Problem with Quadratic and Convex Cost Functions", I.E. Transactions, 5, 314-322.
16. Kuhn, H.W. and Tucker, A.W., (1950), "Nonlinear Programming", Proc. 2nd Berkeley Symp. on Math. Stat. and Prob., Jo Neyman (ed.), Univ. of California Press, Berkeley, Calif., pp. 481-492.
17. Malcolm, D.G., Rosebloom, J.H., Clark, C.E., and Fazar, W., (1959), "Application of a Technique for Research and Development Program Evaluation", Oper. Res., 7(5), 646-669.
18. Salem, A., (1980), "Time Reduction in Activity Networks under Convex Functions", Unpublished Ph.D. dissertation, N.C. State University, Raleigh, N.C.
19. Sigal, C.E., Pritsker, A.A.B., and Solberg, J.J., (1979), "The Use of Cutset in Monte Carlo Analysis of Stochastic Networks", Mathematics and Computers in Simulation, 21, 379-384.
20. Van Slyke, R.M., "Monte Carlo Methods and the PERT Problem", Oper. Res., 11(5), 839-860.

MARKOV MODELS OF MULTI-ECHELON, REPAIRABLE-ITEM
INVENTORY SYSTEMS

by

Donald Gross
Leonidas C. Kioussis
Douglas R. Milier
Richard M. Soland

GWU/IMSE/Serial T-490/84
8 June 1984

The George Washington University
School of Engineering and Applied Science
Department of Operations Research
Institute for Management Science and Engineering

Grant ECS-8200837-01
National Science Foundation
and
Contract N00014-83-K-0217
Office of Naval Research

THE GEORGE WASHINGTON UNIVERSITY
School of Engineering and Applied Science
Department of Operations Research
Institute for Management Science and Engineering

Abstract
of
GWU/IMSE/Serial T-490/84
8 June 1984

MARKOV MODELS OF MULTI-ECHELON, REPAIRABLE-ITEM
INVENTORY SYSTEMS

by

Donald Gross
Leonidas C. Kioussis
Douglas R. Miller
Richard M. Soland

Exact models of finite end-item population, finite repair capacity repairable-item systems are developed using Markov process analyses for both transient and steady state environments. Unlike most currently used multi-echelon models, the infinite population, infinite repair capacity restrictions are removed. Exponential failure and repair times are assumed and the system is modeled as a closed Markovian queuing network.

In the transient case, the finite set of differential equations, and in the steady-state case, the finite set of difference equations, are solved by numerical techniques. The adequacy of these techniques for yielding solutions to *practical* systems is also discussed.

Research Supported by
Grant ECS-8200837-01
National Science Foundation
and
Contract N00014-83-K-0217
Office of Naval Research

THE GEORGE WASHINGTON UNIVERSITY
School of Engineering and Applied Science
Institute for Management Science and Engineering

MARKOV MODELS OF MULTI-ECHELON, REPAIRABLE-ITEM
INVENTORY SYSTEMS

by

Donald Gross
Douglas R. Miller

1. Introduction

Consider a typical multi-echelon repairable-item inventory system as shown schematically in Figure 1. Shown there is a *two* location (bases), *two* level of supply (spares at bases and depot), *two* level of repair (base and depot) system which we shall denote as a (2,2,2) system. The nodes BU_i ($i = 1, 2$) represent operating and spare units (we consider for now only a single item such as a final assembly or a key component) at base i , BR_i ($i = 1, 2$) represent the repair facility at base i , DU represents depot spares, and DR the depot repair facility.

Our goal is to develop exact mathematical models for such finite calling population (finite number of items), finite repair capacity, repairable item provisioning systems in both time-varying and steady-state environments. Specifically, we wish to find the state probability vector (the probability distribution for the system being in its various

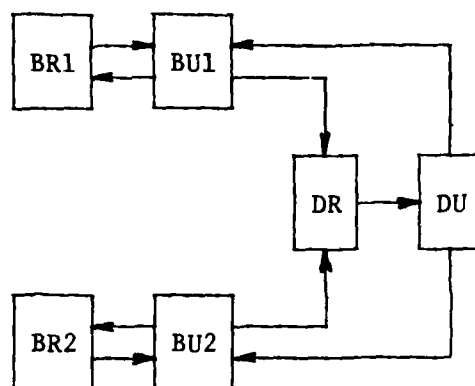


Figure 1. Multi-echelon, repairable item system.

possible states) which will allow us to then calculate measures of performance such as availability (the probability that at least some desirable, prespecified number of components is operational). Ultimately, these models will be used to yield the optimal combination of spares and repair channels at each location in the system.

Assuming times to component failure and component repair times to be exponentially distributed random variables, we have a continuous time Markov process (CTMP). The process is driven by a rate matrix $Q = \{q_{ij}\}$, where q_{ij} is the "rate" of going from state i to state j ; that is, letting $X(t)$ represent the system state at time t ,

$$q_{ij} = \lim_{\Delta t \rightarrow 0} \left[\frac{\Pr\{X(t+\Delta t) = j | X(t) = i\}}{\Delta t} \right], \quad i \neq j;$$

$$q_{ii} = - \sum_{\substack{j \\ (i \neq j)}} q_{ij}.$$

For example, suppose the (2,2,2) system pictured in Figure 1 is in a state (call it i) for which the depot spares pool is not

empty (at least one spare is on hand at the depot). Suppose we consider the event: a component fails at base 1. Describing this state i by the vector $(n_{BU1}, n_{BR1}, n_{BU2}, n_{BR2}, n_{DU}, n_{DR})$, where n_k denotes the number of components at node k in the "network," this event takes the system to a state j , namely, $(n_{BU1}, n_{BR1}, n_{BU2}, n_{BR2}, n_{DU}-1, n_{DR}+1)$, at the rate $q_{ij} = \lambda \alpha_1 n_{BU1}$, where $1/\lambda$ is the mean time to failure (MTTF) of a component and α_1 is the probability (or percentage) of failed items requiring depot repair.

If we denote the state probability (row) vector at time t by $\underline{\pi}(t) = (\pi_1(t), \pi_2(t), \dots, \pi_S(t))$, that is, the i th element, $\pi_i(t)$, is the probability of the system being in state i at time t (there is a finite number of states [call this number S] even though this number can be quite large), then we must solve the finite set of first-order, linear differential (Kolmogorov) equations

$$\underline{\pi}'(t) = \underline{\pi}(t)Q. \quad (1)$$

For steady-state solutions, we are required to solve the finite set of linear algebraic steady state equations,

$$\underline{0} = \underline{\pi}Q, \quad (2)$$

where $\underline{\pi} = (\pi_1, \pi_2, \dots, \pi_S)$ is the steady-state probability vector and $\underline{0}$ is a row vector of all zeroes. In both steady-state and transient cases we have the further condition that the probabilities sum to one, namely,

$$1 = \underline{\pi}(t)\underline{e} = \underline{\pi}\underline{e},$$

where \underline{e} is a column vector, with all components equal to 1.

2. Transient Environment

We are often interested in what happens to such systems in a time-varying environment. For example, a sudden increase in effort (say a peacetime to wartime shift) may cause a sudden decrease in MTTF. In such situations, it is necessary to have $\pi(t)$, and we must solve the finite set of first order linear differential equations given in (1). Except for very small systems (one or two states) analytical techniques such as Laplace transforms are intractable. Since we have a finite set of equations, numerical methods can be employed. Numerical integration schemes such as Runge-Kutta or predictor-corrector methods are possibilities. We choose a different approach, however, which is referred to by some as *randomization*, and has been shown to be more efficient for these kinds of problems [see Arsham, Balana, and Gross (1983) or Grassmann (1977a)]. For details on this technique, which can be derived by a probabilistic argument when viewing the CTMP in a certain way, see Grassmann (1977a and b) or Gross and Miller (1984a and b).

The computational formulas are as follows. Consider a discrete time Markov chain (DTMC) with single-step transition probability matrix

$$P = Q/\Lambda + I,$$

where

$$\Lambda = \max_i |q_{ii}|,$$

that is, Λ is the maximum of the absolute values on the diagonal of the Q matrix. Since a diagonal element of Q is the negative of the sum of the other elements in the row (rows of the Q matrix sum to zero), Λ is actually the absolute value of the minimum (largest

negative) diagonal element of the matrix. This DTMC is referred to as a uniformized embedded DTMC of the CTMP. Denoting by $\phi^{(k)}$ the state probability vector of this DTMC after k transitions, it can be shown (see the above cited references) that

$$\pi_j(t) = \sum_{k=0}^{\infty} \phi_j^{(k)} \frac{(\lambda t)^k e^{-\lambda t}}{k!}.$$

For computational purposes, it is necessary to truncate the infinite sum. The truncation error can be easily bounded since we are discarding a Poisson "tail," so that the computational formula becomes

$$\pi_j(t) = \sum_{k=0}^{T(t,\epsilon)} \phi_j^{(k)} \frac{(\lambda t)^k e^{-\lambda t}}{k!}, \quad (3)$$

where

$$T(t,\epsilon) = \min \left\{ N: \sum_{n=0}^N \frac{e^{-\lambda t} (\lambda t)^n}{n!} > 1 - \epsilon \right\},$$

ϵ being the maximum tolerable error (specified by the user). One advantage of this method over numerical integration is an exact bound on the computational error.

The major computational effort in using (3) is now reduced to finding the state probability vector, $\phi^{(k)}$, of the uniformized embedded DTMC. This can be readily accomplished by the usual recursion,

$$\phi^{(0)} = \pi(0); \quad \phi^{(k+1)} = \phi^{(k)} P. \quad (4)$$

Gross and Miller (1984a) give a more efficient procedure than the successive vector-matrix multiplication of (4), which takes advantage of the sparsity of the P matrix.

3. Steady-state Environment

Solving for the steady-state probability vector π requires solving the set of linear algebraic equations of (2). Since one of these equations is redundant, it is necessary to reduce the equation set by one and use $1 = \pi e$ as the final equation. Thus (2) can be reformulated as

$$\underline{b} = \pi A, \quad (5)$$

where \underline{b} is a vector of all zeroes, except for the last element, which is a 1, and A is the Q matrix with the last column replaced by 1's.

For relatively small systems, the solution can be obtained by inverting A to get

$$\pi = \underline{b} A^{-1}.$$

However, for most realistic problems, the state space (and hence dimension of the A matrix) is too large to obtain A^{-1} efficiently or accurately. This situation suggests iterative procedures such as Jacoby or Gauss-Seidel.

Consider the A matrix as a sum,

$$A = L + D + U,$$

where L is a lower triangular matrix, D is a matrix with only diagonal elements, and U is an upper triangular matrix. Then (5) can be written as

$$\pi(L+D+U) = \underline{b}$$

or

$$\pi D = \underline{b} - \pi(L+U). \quad (6)$$

We can use (6) in an iterative fashion,

$$\pi^{(n+1)} D = \underline{b} - \pi^{(n)}(L+U), \quad (7)$$

where we begin the procedure with some initial guess, say $\underline{\pi}^{(0)}$. This procedure is called Jacoby iteration. Note that in performing the calculations, since D is a diagonal matrix, we compute $\pi_0^{(n+1)}$, $\pi_1^{(n+1)}$, $\pi_2^{(n+1)}$, ... successively. If, as we compute the $\pi_i^{(n+1)}$, we replace the $\pi_i^{(n)}$ on the right-hand side [e.g., in computing $\pi_j^{(n+1)}$, the $\underline{\pi}^{(n)}$ vector is modified to be $\underline{\pi}^{(n)} = (\pi_0^{(n+1)}, \pi_1^{(n+1)}, \dots, \pi_{j-1}^{(n+1)}, \pi_j^{(n)}, \pi_{j+1}^{(n)}, \dots, \pi_N^{(n)})$], this procedure is referred to as Gauss-Seidel iteration, and in matrix representation is

$$(U^T + D)\underline{\pi}^{(n+1)} = \underline{b} - L^T \underline{\pi}^{(n)}, \quad (8)$$

where $\underline{\pi}$ and \underline{b} are now column vectors, and U^T , L^T are the transposes of U and L , respectively.

Two questions remain to be answered concerning use of the iterative procedures of (7) or (8); namely, (i) do the procedures converge, and (ii) when should the iterations be terminated? In general, these procedures may not necessarily converge, although for our well-structured Markov process convergence will take place. The stopping criterion generally used is the Cauchy criterion, namely, stop when

$$\max_i \left| \pi_i^{(n+1)} - \pi_i^{(n)} \right| < \varepsilon_0, \quad (9)$$

where ε_0 is an "arbitrarily" chosen small number. We found using the fractional difference version of (9), namely, stop when

$$\max_i \left| \frac{\pi_i^{(n+1)} - \pi_i^{(n)}}{\pi_i^{(n)}} \right| < \varepsilon_0, \quad (10)$$

to be somewhat more effective. While there has been some success in using Gauss-Seidel (G-S) on Markov models [see Kaufman, Gopinath, and

Wunderlich (1981)], problems exist with respect to rate of convergence and appropriate stopping criteria. The G-S convergence rate can often be improved by using overrelaxation, that is, by weighting with a coefficient greater than one the $\pi_0^{(n+1)}$, $\pi_1^{(n+1)}$, ..., $\pi_{j-1}^{(n+1)}$ used in calculating $\pi_j^{(n+1)}$ [see Kaufman, *et al.* (1981) or Maron (1982)].

Usually, the G-S procedure is applied to a set of equations with a nonsingular matrix (such as A). Consider a nonsingular matrix M with positive diagonal elements and negative off-diagonal elements. The G-S procedure is known to converge for sets of equations with such an M matrix [see Varga (1963)]. Now consider equation set (2), namely,

$$\underline{Q} = \underline{\pi}Q.$$

Multiplying through by -1 gives

$$\underline{Q} = \underline{\pi}[-Q],$$

where $-Q$ has positive diagonal elements and negative off-diagonal elements. However, it is singular, since one equation of this set is redundant. Suppose we arbitrarily set π_S (assuming there are S states) to one, remove the last row of the Q matrix (call this reduced matrix \hat{Q}), and consider solving the reduced $S-1 \times S-1$ set of equations

$$\underline{Q} = \underline{\pi}[-\hat{Q}].$$

Now $-\hat{Q}$ is an M matrix and convergence is guaranteed. Of course the resulting π_i values are relative to $\pi_S = 1$ so that they must be renormalized by dividing each by $\sum_{i=1}^S \pi_i$. How fast convergence takes place still is a key question, however. It turns out [see Kaufman, *et al.* (1981)] that working with the full Q matrix, even though it is singular, speeds convergence, and this is what we also do.

Another procedure is to use the uniformized embedded DTMC of the randomization procedure with transition probability matrix $P = Q/\Lambda + I$. This Markov chain has limiting probabilities given by

$$\phi = \phi P, \quad (11)$$

and they are identical to the π of the CTMP we seek [$\phi = \phi P \Rightarrow \phi = \phi((Q/\Lambda) + I) \Rightarrow \underline{Q} = \phi(Q/\Lambda) \Rightarrow \underline{Q} = \phi Q \equiv \underline{Q} = \pi Q$]. Solving the set of equations given by (11) is no easier, of course, than solving that of (5). However, we know from Markov chain theory that limiting probabilities of a DTMC can be found by iteration, namely,

$$\pi^{(n+1)} = \pi^{(n)} P. \quad (12)$$

Here again, we have computational problems associated with iteration, but we know from Markov chain theory that convergence is guaranteed due to the existence of a steady state vector π (the P matrix is irreducible). The problem of when to stop the iterations remains, however. Using the Cauchy criterion here results in problems similar to those found when using it for G-S iteration, namely, successive probabilities can differ by very small amounts and still be far from the steady state values.

Wallace and Rosenberg (1966) provide a considerably better stopping criterion than the Cauchy criterion of (10). Their stopping rule is based on estimating the rate of convergence by estimating the second eigenvalue of P , and turns out to be: "Stop when

$$\frac{|\pi^{(n+1)} - \pi^{(n)}|}{\left[1 - |\pi^{(n+1)} - \pi^{(n)}|\right]^{1/n}} < \epsilon_0." \quad (13)$$

For details of this development, see Wallace and Rosenberg (1966) or Gross, Kioussis, Miller, and Soland (1984).

4. Results

The following section gives a brief summary of results to date. For greater detail, we refer the reader to Gross and Miller (1984b) and Gross, Kioussis, and Miller (1984) for the transient case and to Gross, Kioussis, Miller, and Soland (1984) for the steady-state case.

4.1 Transient Case

The largest system solved to date using equation (3) directly was a (2,2,2) system (as pictured in Figure 1) with 18 components at base 1 (of which 4 were spares), 13 at base 2 (of which 3 were spares), and 3 spares at the depot. The base repair shops had 2 parallel service channels each, and the depot repair facility had four. This gave a state space of 20,748 ($Q = 20,748 \times 20,748$).

The time-varying environment scenario is shown in Figure 2. At time 6, a shift in MTTF ($1/\lambda$) occurs but it takes until time 10 for the repair facilities to "catch up" in MTTR ($1/\mu$). This simulates a change in usage due to, say, a shift from peacetime to wartime. The measure of effectiveness calculated is the availability at time t ($t = 1, 2, \dots, 15$), where availability is defined as follows:

$$A_1(t) \equiv \Pr\{\text{at least 14 components are operational at base 1 at time } t\}$$

$$A_2(t) \equiv \Pr\{\text{at least 10 components are operational at base 2 at time } t\}$$

$$A_{12}(t) \equiv \Pr\{\text{at least 14 components at base 1 and at least 10 components at base 2 are simultaneously operational at time } t\}.$$

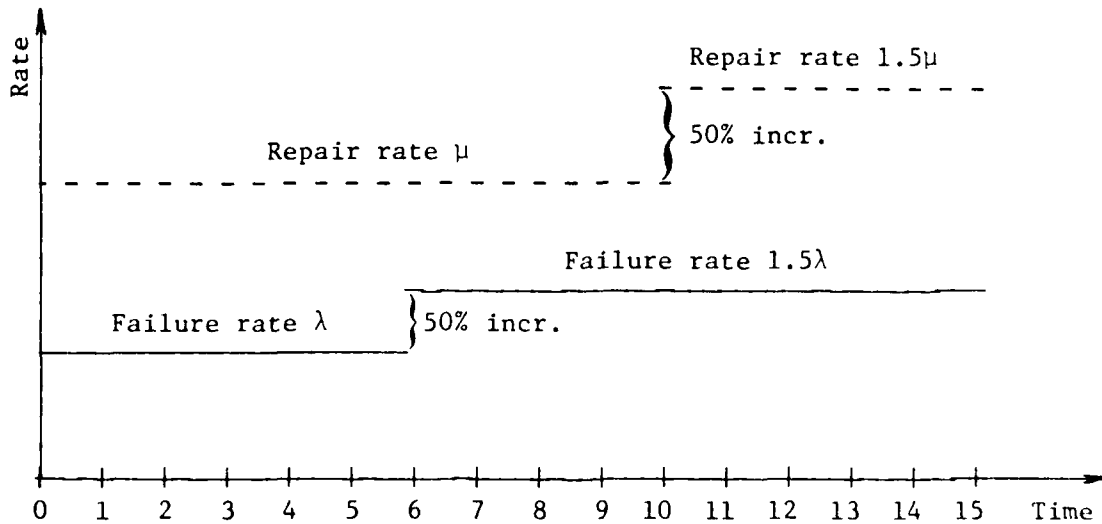


Figure 2. Time-varying environment scenario for sample run.

Figure 3 shows a plot of $A_1(t)$ versus t . Plots of $A_2(t)$ and $A_3(t)$ are similar in nature. The graph shows an initial $A_1(0)$ of 1.0 (we assume at time zero *all* components are operational) and thereafter a drop-off toward the steady-state availability as time increases. At time 6, the increase in failure rate occurs and $A_1(t)$ begins to drop off at a higher rate, heading for a new, lower steady-state availability. However, the increase in repair rate at time 10 causes $A_1(t)$ to begin to rise, heading back toward the original steady-state availability. This run took approximately 25 minutes of CPU time on a VAX 11/780 computer using the randomization computation of (3) with the efficient procedure given in Gross and Miller (1984a) for calculating $\phi^{(k)}$.

As the systems become more complex (more bases, multiple component types, indenture, more echelons, etc.) the state-space grows rapidly. We have solved a problem with three bases, yielding a state-space of size 43,278,703, by truncating the state-space ("lumping" low probability states into several absorbing states resulting in a truncated state space

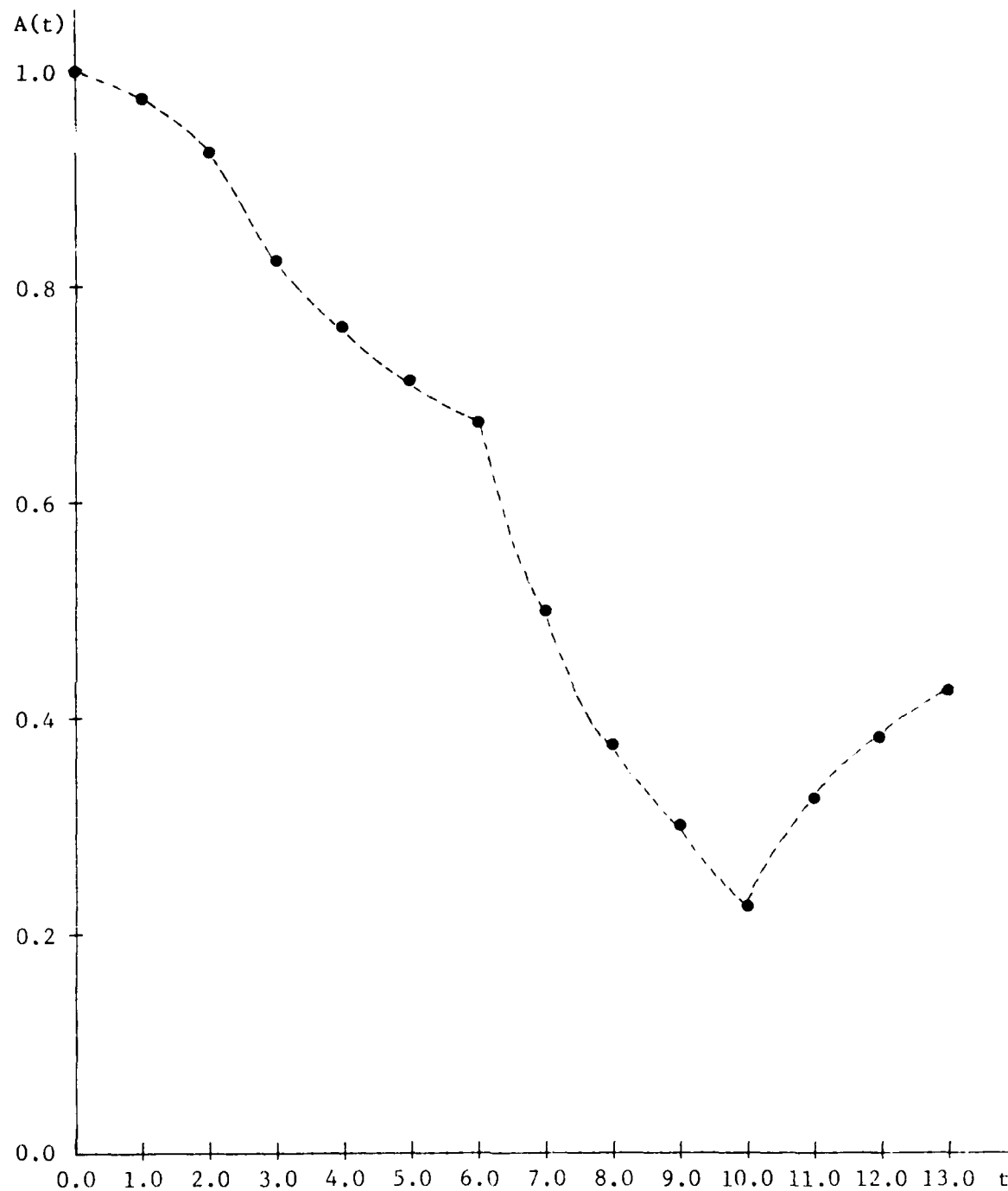


Figure 3. $A_1(t)$ versus t for sample run.

of approximately 15,000 states) on the VAX 11/780 in approximately 30 minutes [see Gross, Kiuoussis, and Miller (1984)].

4.2 Steady-state Case

Ironically, computational success has been far more elusive for the steady-state situation than for the transient case. The problem is the stopping criterion for these iterative procedures (a problem not present when dealing with transient solutions). In the transient case, the randomization procedure guarantees an accuracy to within a pre-specified ϵ . For steady state, using either the Cauchy or the Wallace-Rosenberg stopping rule does not guarantee errors within ϵ . Table 1 shows some computations for a (1,1,1) system which is the classical machine repair with spares model of queueing theory. For this model the availability can be computed analytically, which allowed us to estimate the actual error. The columns under P-WR show the results of using (12) with the stopping criterion of (13), the Wallace-Rosenberg approach, while the GS-C columns show results for (8) with the stopping criterion of (10), the Gauss-Seidel approach.

The circled elements show the cases for which the error specification, ϵ , was exceeded. While there were more cases of exceeding the stopping rule error specification in P-WR, the error excesses were larger, especially for the larger population cases, under GS-C. But GS-C stopped in far fewer iterations in almost all cases (except for the very small population cases), and it is the number of iterations that consumes most of the CPU time.

The last column shows a rerun of GS-C, ignoring the stopping criterion and performing the same number of iterations as used for the P-WR procedure. The errors essentially went to zero, which indicates that if a better stopping criterion could be found, Gauss-Seidel iteration

TABLE 1
RESULTS FOR (1,1,1) MODEL

Desired # Operating (M)	# Spares (Y)	# Repair Channels (C)	$\rho \equiv$ M\lambda/C\mu	Exact Avail*	P-WR		GS-C		GS-WR Error
					Error in # Avail.	Iter	Error in # Avail.	Iter	
5	1	1	1.5	.2123	.0071	8	.0019	11	
			1.0	.4434	<u>.0123</u>	19	.0023	12	
			0.5	.7951	.0044	20	.0005	12	
10	5	3	1.5	.1533	<u>.0137</u>	36	.0030	20	
			1.0	.5719	<u>.0122</u>	64	<u>.0136</u>	30	.0001
			0.5	.9745	.0017	55	.0004	32	
20	4	5	1.5	.0386	.0024	57	.0032	42	
			1.0	.3217	<u>.0113</u>	98	.0072	52	.0000
			0.5	.8790	.0034	71	.0004	46	
50	5	3	1.5	.0039	.0003	183	.0006	94	
			1.0	.3640	<u>.0146</u>	313	<u>.0157</u>	148	.0000
			0.5	.9714	.0017	156	.0000	117	
100	10	3	1.5	.0001	.0000	467	.0000	207	
			1.0	.4475	<u>.0152</u>	789	<u>.0385</u>	317	.0000
			0.5	.9991	.0003	302	.0000	241	
100	20	5	1.0	.6025	<u>.0125</u>	1100	<u>.0954</u>	321	.0001

*Percentage of time the population is operating at full strength M.

might be a viable approach. Runs for some (2,2,2) systems and more detailed discussion of these steady-state procedures can be found in Gross, Kioussis, Miller, and Soland (1984).

References

- Arsham, H., A. R. Balana, and D. Gross (1983). Numerical solutions of machine repair problems, *Computers & Industrial Engineering*, 7, 149-157.
- Grassmann, W. (1977a). Transient solutions in Markovian queueing systems, *Computers & Operations Research*, 4, 47-56.
- Grassmann, W. (1977b). Transient solutions in Markovian queues, *European Journal of Operational Research*, 1, 396-492.
- Gross, D. and D. R. Miller (1984a). The randomization technique as a modelling tool and solution procedure for transient Markov processes, *Operations Research*, 32, 343-361.
- Gross, D. and D. R. Miller (1984b). Multi-echelon repairable item provisioning in a time-varying environment using the randomization technique, *Naval Research Logistics Quarterly*, in press.
- Gross, D., L. C. Kioussis, and D. R. Miller (1984). Transient behavior of large Markovian multi-echelon repairable item inventory systems using a truncated state-space approach, Technical Paper GWU/IMSE/Serial T-488/84, Institute for Management Science and Engineering, George Washington University
- Gross, D., L. C. Kioussis, D. R. Miller, and R. M. Soland (1984). Computational aspects for determining steady state availability in Markovian multi-echelon repairable item inventory models, Technical Paper GWU/IMSE/Serial T-483/84, Institute for Management Science and Engineering, George Washington University.

- Kaufman, L., G. Gopinath, and E. F. Wunderlich (1981). Analysis of packet network congestion control using sparse matrix algorithms, *IEEE Transactions on Communications*, COM-29, 453-464.
- Maron, M. J. (1982). *Numerical Analysis*. Macmillan, New York.
- Varga, R. S. (1963). *Matrix Iterative Analysis*. Prentice-Hall, Englewood Cliffs, New Jersey.
- Wallace, V. L. and R. S. Rosenberg (1966). RQA-1, the recursive queue analyzer, Systems Engineering Laboratory Report 97742-1-T, Department of Electrical Engineering, University of Michigan, Ann Arbor.

STUDYING MODEL ASSUMPTIONS
IN PROCESS QUALITY CONTROL

C. P. Quesenberry
North Carolina State University
Raleigh, NC 27695-8203

1. INTRODUCTION

In process quality control by variables we typically assume that a sequence of observations $X_1, X_2, \dots, X_n, \dots$ is observed in time. The X_i 's themselves are often batch means. The basic model usually assumes that the X_i 's are independently and identically distributed normal random variables, i.e., are i.i.d. $N(\mu, \sigma^2)$ r.v.'s. Some of the most widely used methods of process quality control, such as Shewhart charts for sample means and ranges and CUSUM tests for sample means, are designed to detect changes in the process that cause either μ or σ to shift to different values. However, the observed data contain much information that can be used for purposes other than detecting changes in the mean and variance of an assumed normal model. In this talk, we shall propose computing certain statistics which capture all of the information in the data, beyond that in the sample mean and variance, and consider methods of using these statistics to detect a number of types of violations of the basic model assumptions.

The statistics which we propose to compute are called sequential uniform residuals, due to the fact that they have known uniform distributions when the normal model is correct. These residuals are not new, but have been derived and studied by the present speaker and co-workers in earlier work, however, they have not been considered in the context of process quality control. We feel that they have excellent potential for many useful applications in this area.

2. UNIFORM AND NU RESIDUALS

Let $X_1, X_2, \dots, X_n, \dots$ denote a sequence of random variables, and when these are i.i.d. $N(\mu, \sigma^2)$ random variables we shall say that normal model assumptions hold. We define the following (sequential) sample quantities:

$$\bar{X}_r = \sum_1^r X_j / r, \quad S_r^2 = \sum_1^r (X_j - \bar{X}_r)^2 / r,$$

(to compute these sequentially, see Youngs and Cramer, 1972)

$$\bar{X}_r = [(r-1)\bar{X}_{r-1} + X_r] / r, \quad S_r^2 = \left(\frac{r-1}{r}\right) S_{r-1}^2 + \left(\frac{1}{r}\right) (X_r - \bar{X}_r)^2$$

$$A_r = [r(r-2)]^{1/2} (X_r - \bar{X}_r) / (r-1) S_{r-1}$$

$$u_{r-2} = G_{r-2}(A_r); \quad r = 2, 3, \dots, n, \dots$$

When $G_v(\cdot)$ is a Student-t distribution function with v degrees of freedom.

If the normal model assumptions hold, then the values u_1, u_2, \dots were shown in O'Reilly and Quesenberry (1973), O-Q, to be i.i.d. uniform random variables on the unit interval. A careful examination of the value A_r above shows that it is a Studentized value of the residual of the r^{th} observed value from the mean of the first r values. For this reason, we shall call the value u_r a uniform residual. Our purpose in this paper is to consider using these uniform residuals and other related quantities in methods for studying the model assumptions in statistical process control.

These uniform residuals were first derived in O-Q by the method of conditional probability integral transformations introduced by those authors. These residuals can also be obtained as a special case of the uniform residuals from regression models with normal error structure as given in O-Q, and considered also in Quesenberry (1983) and Hester and Quesenberry (1984). In the present work, we shall discuss some ways in

which these sequential uniform residuals can be used in statistical process control, however, we first set out some basic properties of these statistics which, we feel, motivate the use of these quantities. When normal model assumptions hold, the uniform residuals have the following properties:

Property 1. The quantities u_1, u_2, \dots, u_N , are i.i.d. uniform random variables on the unit interval $(0,1)$. (O-Q, 1973).

Property 2. The vector $u_r = (u_1, \dots, u_r)'$ is independent of the vectors $\bar{X}_r = (\bar{X}_3, \dots, \bar{X}_r)'$ and $S_r^2 = (S_2^2, \dots, S_r^2)'$ for all $r = 2, 3, \dots$. (This follows from the completeness and sufficiency of $T_r = (\bar{X}_r, S_r^2)'$, and a well-known result of Basu (1955).)

Property 3. The uniform residuals vector u_r is a maximal invariant with respect to linear transformations of the data. (Quesenberry and Starbuck (1976), Q-S).

Property 4. If a most powerful similar or a most powerful invariant (re to linear transformations) test exists for testing the normal model against any particular alternative, then a test of equal power can be based on the uniform residuals. (Q-S).

Let $\phi(\cdot)$ denote the distribution function of a $N(0,1)$ distribution, and $\phi^{-1}(\cdot)$ its inverse function. Then we define NU (normal-uniform) residuals

$$z_j = \phi^{-1}(u_j), \quad j = 1, 2, \dots$$

and z_1, z_2, \dots are i.i.d $N(0,1)$ r.v.'s, when the normal model holds.

For some problems, there are advantages in considering the NU residuals rather than, or in addition to, the uniform residuals.

3. STUDYING THE RESIDUALS

In view of the properties of the sequential uniform residuals discussed in section 2, the principal task is to find effective ways to study these values in order to detect deviations from model assumptions. We can consider two basic settings, which present somewhat different problems. First, we can consider the analysis of residuals for a set of past data, and, second, we can consider the sequential analysis of data as it is observed in time. In each of these cases, we may wish to use the data to attempt to identify different types of model misspecification which can occur. There are many different types of problems which can be considered and we must limit our discussion here to a few particular problems. In this section we consider the simple but useful method of graphing the u 's in order to look for patterns, and then suggest tests for uniformity on the uniform residuals and of normality on the NU residuals as general methods of analysis to detect anomalies in the data. In the next section, we suggest using these residuals to detect outliers.

We consider first some methods for analyzing the uniform and NU residuals that are designed to detect a wide range of deviations from the basic normal model. Since these methods are expected to perform reasonably well in detecting a large class of alternatives, we cannot expect them to be most effective against particular restricted alternatives. If we wish to focus on a particular alternative, then it may be possible to find a test or other analysis technique which is especially sensitive for detecting it. In this section, we shall consider analyses based on plotting the residuals and computing some omnibus goodness-of-fit tests.

Graphing Techniques

As a first analysis, we suggest plotting the uniform (or NU) residuals against the index and studying these plots for trends. Due to the recursive property of these residuals, the plots can be made sequentially when each observation is taken, in order to identify problems as early as possible. Under the null hypothesis, the uniform residuals should tend to form a uniform band between the lines $u = 0$ and $u = 1$. The possible types of patterns that indicate model specification errors is large. Indeed, any recognizable pattern among these points will likely require further study. The type of misspecification that leads to a particular point pattern can sometimes be deduced by recalling the nature of the transformations in (2.1) viz., that A_j is a Studentized residual and u_j is obtained by transforming A_j with the appropriate Student-t distribution function.

The uniform and NU residuals are equivalent statistics, and contain the same information; however, some patterns or anomalies will be more apparent in the graphs of one or the other of these types of residuals. One instance of this is in detecting outliers. The plots of NU residuals will display outliers more clearly than will the plots of uniform residuals. The detection of outliers will be considered in Section 4.

An Omnibus Test for Uniformity

One type of test statistic which we shall often want to compute is an omnibus test of simple uniformity on the values of μ . Such omnibus tests have reasonably good power against a wide range of alternatives. There are a large number of omnibus goodness-of-fit tests which can be used to make tests for uniformity. Reasonably extensive power studies of tests for uniformity have been made by Quesenberry and Miller (1977), Q-M, and by Miller and Quesenberry (1979), M-Q. These papers review much of the literature in this area of goodness-of-fit testing. Based upon the results in these papers, we recommend the Neyman smooth test.

Neyman (1937) posed a statistic designed to have high power for testing uniformity against certain classes of alternative distributions on the unit interval. (See M-Q and Kendall and Stuart (1961), p. 444.) The test is defined as follows. The Legendre polynomials π_r are given by; for $r = 0, 1, 2, 3, 4$; and $0 \leq y \leq 1$;

$$\begin{aligned}\pi_0(y) &= 1, \pi_1(y) = \sqrt{12}(y - \frac{1}{2}), \\ \pi_2(y) &= \sqrt{5}[6(y - \frac{1}{2})^2 - \frac{1}{2}], \pi_3(y) = \sqrt{7}[20(y - \frac{1}{2})^3 - 3(y - \frac{1}{2})], \\ \pi_4(y) &= 210(y - \frac{1}{2})^4 - 45(y - \frac{1}{2})^2 + 9/8.\end{aligned}\tag{3.1}$$

Then put

$$t_r = \sum_{j=1}^N \pi_r(u_j) \text{ for } r = 1, 2, 3, 4,$$

and the Neyman smooth test rejects for large values of the statistic

$$p_4^2 = (1/N) (t_1^2 + t_2^2 + t_3^2 + t_4^2).\tag{3.2}$$

Neyman showed that p_4^2 has a limiting $\chi^2(4)$ distribution when u_1, \dots, u_N are i.i.d. uniform r.v.'s. Computations of upper .1, .05, and .01 percentage points in M-Q indicate that this approximation is reasonably good for N as small as ten, or even smaller in some cases. This approximation is particularly convenient because it can be used to determine the observed significance level or p-value of the test for uniformity. Thus, in order to obtain an overall assessment of the validity of the normal model, we compute the u_j 's from (2.1), the value of p_4^2 from (3.2), and then evaluate

$$NS4_PV = \text{p-value} = P\{\chi^2(4) > p_4^2\}.\tag{3.3}$$

For $0 < \alpha < 1$, if $NS4_PV \leq \alpha$, then, of course, we reject the normal model at the α level. In practice, we compute $NS4_PV$ and view it as a general coefficient of validity of the normal model.

Tests of Normality on NU Residuals

In addition to (or in place of) omnibus tests for uniformity, we can test the normality of the NU residuals z_1, \dots, z_N , defined at the end of Section 2. There are many good tests of normality available today, and there is no strong reason to favor a particular one. We slightly prefer the Anderson-Darling test. Two points should be noted in this context. Although z_1, \dots, z_N ($N = n-2$) are i.i.d. $N(0,1)$ when the normal assumptions hold, it has been shown by Stephens (1974) and independently by Dyer (1974) that statistics for testing composite normality often have better power for testing simple normality than tests designed to test the simple normality null hypothesis. Another point that should be noted is that the most popular tests for normality do not have solved distribution theory that allows the exact determination of p-values of the tests. We feel this is a considerable disadvantage for these tests.

4. DETECTING OUTLIERS

In many process control problems, an occasional observation will appear which is either much larger or smaller than its fellows. The question then is how one is to decide when an observation is an "outlier" and when it is a feasible value under the normal model assumptions. The exact distribution theory of uniform residuals provides an especially simple and elegant solution for the problem of detecting outliers.

We shall declare an observation a left outlier if it is too small and thus its uniform residual is too near zero, and we call it a right outlier if it is too large and its uniform residual is thus too near one. In view of the nature of sequential uniform residuals, this means that an observation is called a left outlier if it is too small when compared with the observations preceding it, and, similarly, it is a right outlier if it is

too large in comparison with the values preceding it. Thus these might be called sequential outliers.

Suppose that we are willing to incorrectly decide that observations are too large and that they are too small each at the rate of 1 in N_0 observations, when the normal model is correct. Then we apply the following rejection (identification) rule:

Declare x_r a left outlier if $u_{r-2} < 1/N_0$,

Declare x_r a right outlier if $u_{r-2} > (N_0-1)/N_0$. (4.1)

The overall rejection rate is $2/N_0$, when the model is correct. This is a reasonable procedure for screening observations as they arrive sequentially. This procedure is, of course, equivalent to rejecting individual observations as outliers if they fall outside the lines $u = 1/N_0$ and $u = (N_0-1)/N_0$ on the graph discussed in Section 3.

If we wish to decide if a sequence u_1, \dots, u_N of residuals from past data contains outliers we can apply the rule above to perform tests, or we can appeal directly to the distribution theory for order statistics from a uniform distribution. Let $u_{(1)}$ and $u_{(N)}$ denote the smallest and largest values among the residuals, respectively. Moreover, let P_L denote the p-value for testing that the point associated with $u_{(1)}$ is a left outlier, and P_R denote the p-value for testing that the point corresponding to $u_{(N)}$ is a right outlier. When the normal model holds these values are given by

$$P_L = 1 - (1 - u_{(1)})^N, \quad P_R = 1 - u_{(N)}^N. \quad (4.2)$$

The use of these formulas will be illustrated with numerical examples in the following section.

5. NUMERICAL EXAMPLES

To illustrate the techniques discussed above, we have computed the uniform residuals for a number of data sets. A random sample of size 50 was generated from each of four distributions and the uniform residuals were plotted against the index. Samples were drawn from a normal, exponential, uniform and Cauchy distribution. The graphs for these samples are given in the following Figures 1 - 4. The p-values NS^4_PV , P_L and P_R are given in Table 1.

The plot of the uniform residuals for the normal sample in Figure 1 shows no anomalous patterns and the p-values are easily in the acceptance range.

The graph for the exponential sample in Figure 2 does show important patterns that indicate a nonnormal distribution. There are no observations very near zero - which is a reflection of the fact that the normal density is positive on the negative reals but the exponential density is zero on the negative reals. Note that the p-values are all suspect. The Neyman smooth statistic p-value is 0.01386, $P_L = 0.99981$ is too large, and $P_L = 0.02536$.

The analysis of the uniform sample shows a value of $NS^4_PV = 0.06839$, which is suspect, and a value of P_R that is too large, again. The graph in Figure 3 shows no points very near zero or one, which is a reflection of the fact that the uniform density has thinner tails than the normal density.

The Cauchy sample is easily rejected by the goodness-of-fit test, and its tendency to throw outliers is evident in the p-values of the order statistics.

Finally, we computed the uniform residuals and the p-values for the

data given by Ott (1975). Note that in Figure 5 for this data the points display a rising trend beginning at about the 76th or 77th original data points. This is due to a trend in the data discussed by Ott. Also, both P_L and P_R are significant at the 0.05 level.

FIGURE 1: PLOT FOR NORMAL SAMPLE

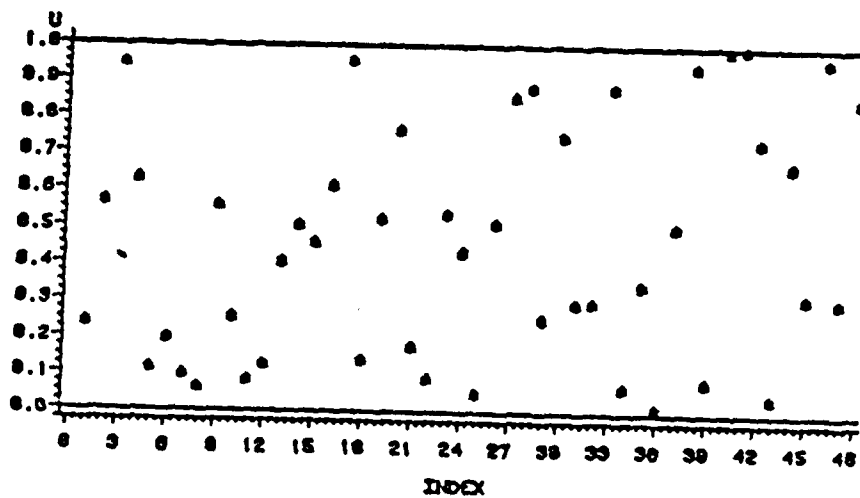


FIGURE 2: PLOT FOR EXPONENTIAL SAMPLE

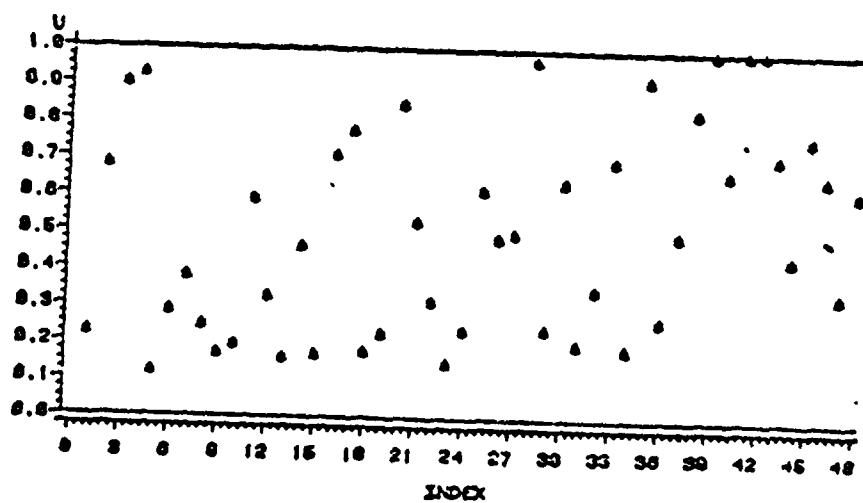


FIGURE 3: PLOT FOR UNIFORM SAMPLE

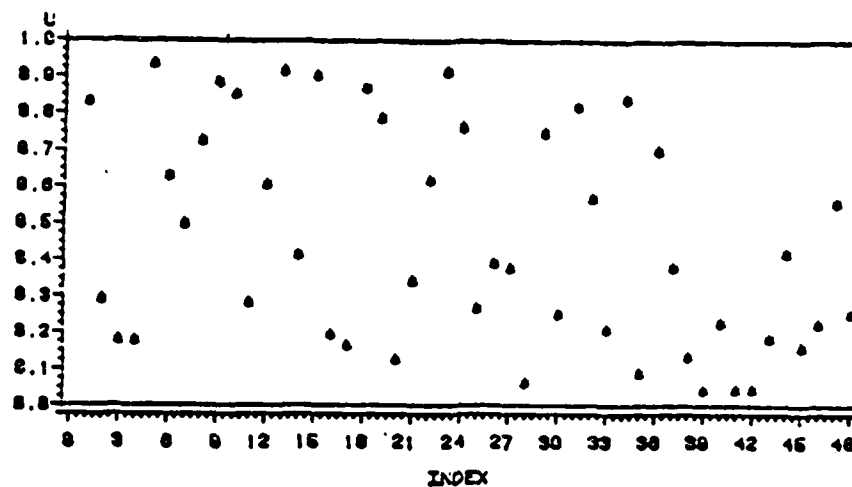


FIGURE 4: PLOT FOR CAUCHY SAMPLE

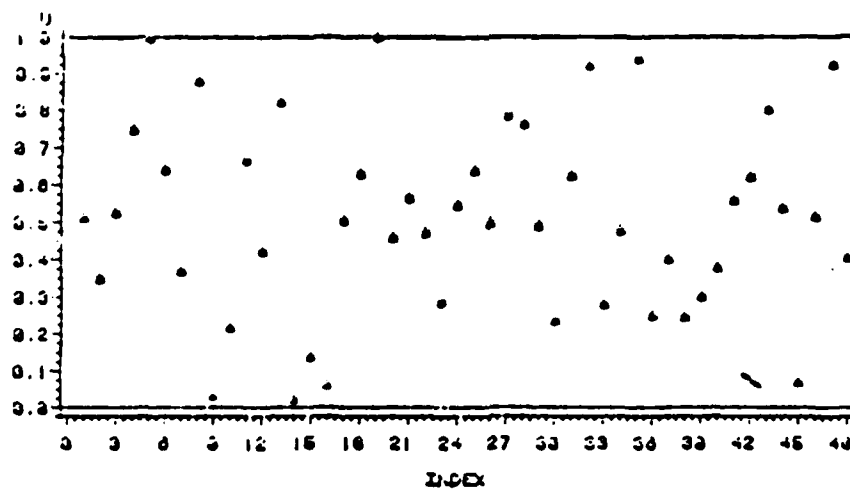


FIGURE 5: PLOT FOR OTT DATA

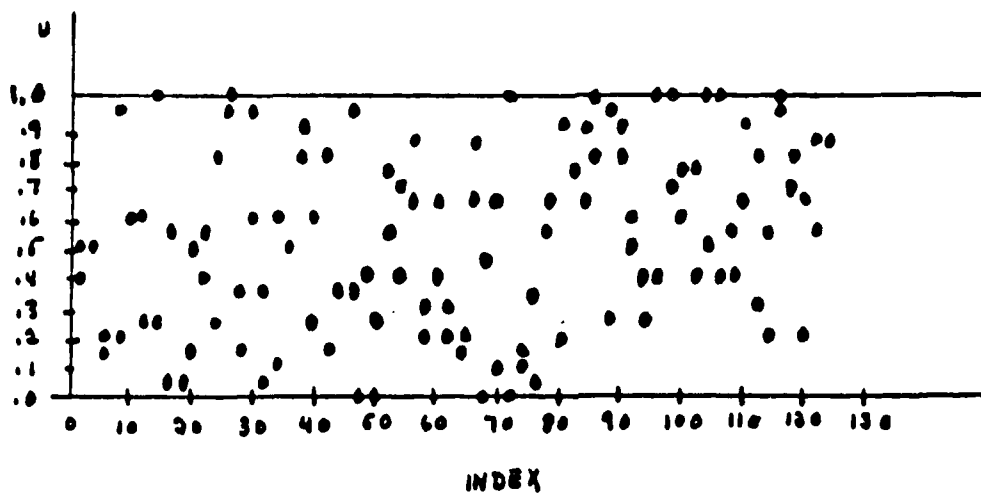


TABLE 1: GOODNESS OF FIT AND OUTLIERS P-VALUES

Parent Distribution	NS4_PV	P_L	P_R
Normal	0.45931	0.93523	0.49131
Exponential	0.01386	0.99981	0.02536
Uniform	0.06839	0.50572	0.98129
Cauchy	7.95071E-08	2.06501E-14	0.021174
Ott	0.039851	0.014772	0.04294

REFERENCES

- Basu, D. (1955), "On Statistics Independent of a Complete Sufficient Statistic" Sankhya, 15, 377-380.
- Dyer, A. R. (1974), "Hypothesis Testing Procedures for Separate Families of Hypotheses," Journal of the American Statistical Association, 69, 140-145.
- Hester, Allen and C. P. Quesenberry (1984), Analyzing Uniform Residuals for Heteroscedasticity (Unpublished manuscript).
- Miller, F. L., Jr., and Quesenberry, C. P. (1979), "Power Studies of Tests for Uniformity," Communications in Statistics - Simulation and Computations, B8(3), 271-290.
- Neyman, J. (1937), "'Smooth' Test for Goodness of Fit," Skandinavisk Aktuarietidskrift, 20, 149-199.
- Ott, E. R. (1975), Process Quality Control, McGraw-Hill Book Company.
- O'Reilly, F. J., and Quesenberry, C. P. (1973), "The Conditional Probability Integral Transformation and Applications to Obtain Composite Chi-Square Goodness-of-Fit Tests," Annals of Statistics, 1, 74-83.
- Quesenberry, C. P. (1983), Analyzing Uniform Residuals for Time Changes in Regression Models (Unpublished manuscript).
- Quesenberry, C. P., and Miller, F. L., Jr. (1977), "Power Studies of Some Tests for Uniformity," Journal of Statistical Computation and Simulation, 5, 169-191.
- Quesenberry, C. P., and Starbuck, R. R. (1976), "On Optimal Tests for Separate Hypotheses and Conditional Probability Integral Transformations," Communications in Statistics, A5(6), 507-524.
- Stephens, M. A. (1974), "EDF Statistics for Goodness of Fit and Some Comparisons," Journal of the American Statistical Association, 69, 730-737.
- Youngs, E. A., and Cramer, E. M. (1971). "Some Results relevant to Choice of Sum and Sum-of-Product Algorithms," Technometrics, 13, 657-665.

396

(BLANK)

Recent Research in Experimental Design for Quality Improvement
with Applications to Logistics

George E. P. Box

1. LOGISTICS AND QUALITY CONTROL

Important measures of military competence such as performance capability and readiness rate are greatly influenced by the quality of the weapons and of the other devices available to the soldier.

A traditional philosophy of quality control has been to "inspect bad quality out" and indeed there are famous military standards that employ this philosophy. W. Edwards Deming (1982) has likened this to making toast according to the recipe "you burn it and I'll scrape it", and has urged the alternative philosophy of assuring that good quality has been built in to the product in the first place. In particular he attributes to the latter philosophy the success of Japanese industry in producing high quality products at low cost. A typical example of the dramatic consequences that have been attributed to these differences of approach are the air-conditioner defect rates shown in Table 1 and quoted by David Garvin (1983).

(In the factory: Assembly line defects per 100 units)			
		American	Japanese
Total		63.5	0.95
Leaks		3.1	0.12
Electrical		3.3	0.12

(In the field: Service call rate per 100 units under first year warranty coverage)			
		American	Japanese
Total		10.5	0.6
Compressors		1.0	0.05
Thermostats		1.4	0.002
Fan motors		0.5	0.028

TABLE 1. Defect rates in US and Japanese air conditioners

Similar comparisons have been made between defect rates in American and Japanese automobiles.

The same United States industry that makes air conditioners and motor vehicles also makes military hardware. It seems clear therefore that a major change in quality philosophy could produce a major improvement in the reliability of the Army's equipment. The philosophy of "building quality in" employs a policy of never ending quality improvement which may be typified in

terms of the traditional statistical model

$$y = f(\tilde{x}_1) + e$$

where y is a quality characteristic believed to depend on a set of variables denoted by \tilde{x}_1 whose identity is known, and e is the difference $y - f(\tilde{x}_1)$ usually referred to as error. (Such "errors" are often somewhat arbitrarily imbued by the theoretician with properties of randomness, normality independence and homoscedasticity). In reality e is a function $e(\tilde{x}_2)$ of a number of additional variables, \tilde{x}_2 say, which affect the system but whose identity is usually unknown. In general, quality improvement is achieved by transferring elements of the unknown factor vector \tilde{x}_2 into the known factor vector \tilde{x}_1 as indicated below

$$y = f(\underset{\text{known}}{\tilde{x}_1}) + e(\underset{\text{unknown}}{\tilde{x}_2})$$

The effect of such transfer is two-fold

- (i) to reveal effects of previously unknown factors which may then be adjusted to levels yielding higher quality and/or used to control the process.
- (ii) to remove variation previously caused by haphazard changes in these factors.

Some of the statistical techniques which contribute to this transfer are quality control charting (including Shewhart, Cusum, Pareto and Fishbone charts) and designed experimentation on line and off line (employing in different and appropriate contexts factorial, fractional factorial and orthogonal array designs, evolutionary operation and response surface methods).

2. SCIENTIFIC METHOD AND QUALITY

Charting and experimentation are examples respectively of passive surveillance and active intervention both of which are important elements in scientific method which it is desirable to consider further.

Humans differ from other animals most remarkably in their ability to learn. It is clear that although throughout the history of mankind technological learning has taken place, until three or four hundred years ago change occurred very slowly. One reason for this was that in order to learn something - for example, how to make fire or champagne - two rare events needed to coincide: (a) an informative event had to occur, and (b) a person able to draw logical conclusions and to act on them had to be aware of that informative event.

Passive surveillance is a way of increasing the probability that the rare informative event will be constructively taken note of and is exemplified by quality charting methods. Thus a Shewhart chart is a means to ensure that

possibly informative events are brought to the attention of those who may be able to discover in them an "assignable cause" (Shewhart 1931) and act appropriately.

Active intervention by experimentation aims, in addition, to increase the probability of an informative event actually occurring. A designed experiment conducted by a qualified experimenter can dramatically increase the probability of learning because it increases simultaneously the probability of an informative event occurring and also the probability of the event being constructively witnessed. Recently there has been much use of experimental design in Japanese industry particularly by Genichi Taguchi (Taguchi and Wu (1980)) and his followers. In off-line experimentation he has in particular emphasized the use of highly fractionated designs and orthogonal arrays and the minimization of variance.

In the remainder of this paper we briefly outline some recent research on the use of experimental design in the improvement of quality.

3. USE OF SCREENING DESIGNS TO IMPROVE QUALITY

Table 2 shows in summary a highly fractionated two-level factorial design employed* as a screening design in an off-line welding experiment performed by the National Railway Corporation of Japan (Taguchi and Wu, 1980). In the column to the right of the table is shown the observed tensile strength of the weld, one of several quality characteristics measured.

The design was chosen on the assumption that in addition to main effects only the two-factor interactions AC, AG, AH, and GH were expected to be present. On that supposition, all nine main effects and the four selected two-factor interactions can be separately estimated by appropriate orthogonal contrasts, the two remaining contrasts corresponding to the columns labelled e_1 and e_2 measure only experimental error. Below the table are shown the grand average, the fifteen effect contrasts, and the effects plotted on a dot diagram. When the effects are plotted on normal probability paper, thirteen of them plot roughly as a straight line but the remaining two, corresponding to the main effects for factors B and C, fall markedly off the line, suggesting that over the ranges studied, only factors B and C affect tensile location by amounts not readily attributed to noise.

If this conjecture is true, then, at least approximately, the sixteen runs could be regarded as four replications of a 2^2 factorial design in factors B and C only. However, when the results are plotted in Figure 1 so as to reflect this, inspection suggests the existence of a dramatic effect of a different kind - when factor C is at its plus level the spread of the

* To facilitate later discussion we have set out the design and labelled the levels somewhat differently from Taguchi.

A: Kind of Welding Rods
 B: Period of Drying
 C: Welded Material
 D: Thickness
 E: Angle
 F: Opening
 G: Current
 H: Welding Method
 J: Preheating

Factor	Column	Number	0	1	2	3	e ₁	4	G	5	F	6	GH	7	AC	A	8	E	9	AH	10	e ₁	11	AG	12	J	13	B	14	C	15	Tensile strength kg/mm ²
Run		1	+	-	-	+	+	-	-	+	+	+	+	-	-	-	-	+	+	+	+	-	+	+	-	-	-	-	-	+	+	43.7
		2	+	+	-	-	-	-	-	-	-	+	+	+	+	-	-	-	-	+	+	+	+	+	+	+	+	-	-	-	-	40.0
		3	+	-	+	+	-	-	-	+	+	-	-	+	+	-	-	+	+	-	-	+	+	+	+	-	-	+	+	-	-	42.4
		4	+	+	+	+	+	-	-	-	-	-	-	-	+	-	-	-	-	+	+	-	-	+	+	+	+	+	+	+	+	44.7
		5	+	-	-	+	+	+	+	-	-	-	-	+	+	-	-	+	+	+	+	-	-	-	-	+	+	+	+	-	-	42.4
		6	+	+	-	-	-	+	+	+	+	-	-	-	-	-	-	-	-	+	+	+	+	-	-	-	-	+	+	+	+	45.9
		7	+	+	+	+	-	+	+	-	-	+	+	+	+	-	-	+	+	-	-	+	+	-	-	+	+	-	-	+	+	42.2
		8	+	+	+	+	+	+	+	+	+	+	+	+	+	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	40.6
		9	+	-	-	+	+	-	-	+	+	+	+	-	-	+	+	-	-	-	-	+	+	-	-	+	+	+	+	-	-	42.4
		10	+	+	-	-	-	-	-	-	-	+	+	+	+	+	+	+	+	-	-	-	-	-	-	-	-	+	+	+	+	45.5
		11	+	+	+	+	-	-	-	+	+	-	-	+	+	+	+	+	+	+	+	+	+	+	+	+	+	-	-	+	+	43.6
		12	+	+	+	+	+	+	+	-	-	-	-	-	-	+	+	+	+	+	+	+	+	+	+	-	-	-	-	-	-	40.6
		13	+	+	-	-	-	+	+	-	-	-	-	-	+	+	+	+	+	-	-	+	+	+	+	-	-	-	-	+	+	44.0
		14	+	+	-	-	-	+	+	+	+	+	+	-	-	+	+	+	+	+	-	-	-	+	+	+	+	-	-	-	-	40.2
		15	+	-	+	+	-	+	+	-	-	+	+	-	-	+	+	-	-	+	+	-	-	+	+	-	-	+	+	-	-	42.5
		16	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	46.5
Effect			43.0	.13	-.15	-.30	-.15	-.15	-.15	.40	.40	-.03	.38	.40	.40	.40	.40	-.05	.43	.13	.13	.13	.13	.13	-.38	2.15	3.10					

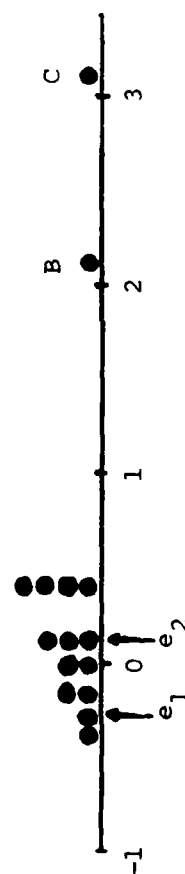


TABLE 2. A fractional two-level design used in a welding experiment showing observed tensile strength and effects. Below the estimated effects are plotted as a dot diagram.

data appears much larger* than when it is at its minus level. Thus, in addition to detecting shifts in location due to B and C, the experiment may also have detected what we will call a dispersion effect due to C. The example raises the general possibility pursued in the remainder of this paper of analyzing unreplicated designs for dispersion effects as well as for the more usual location effects.

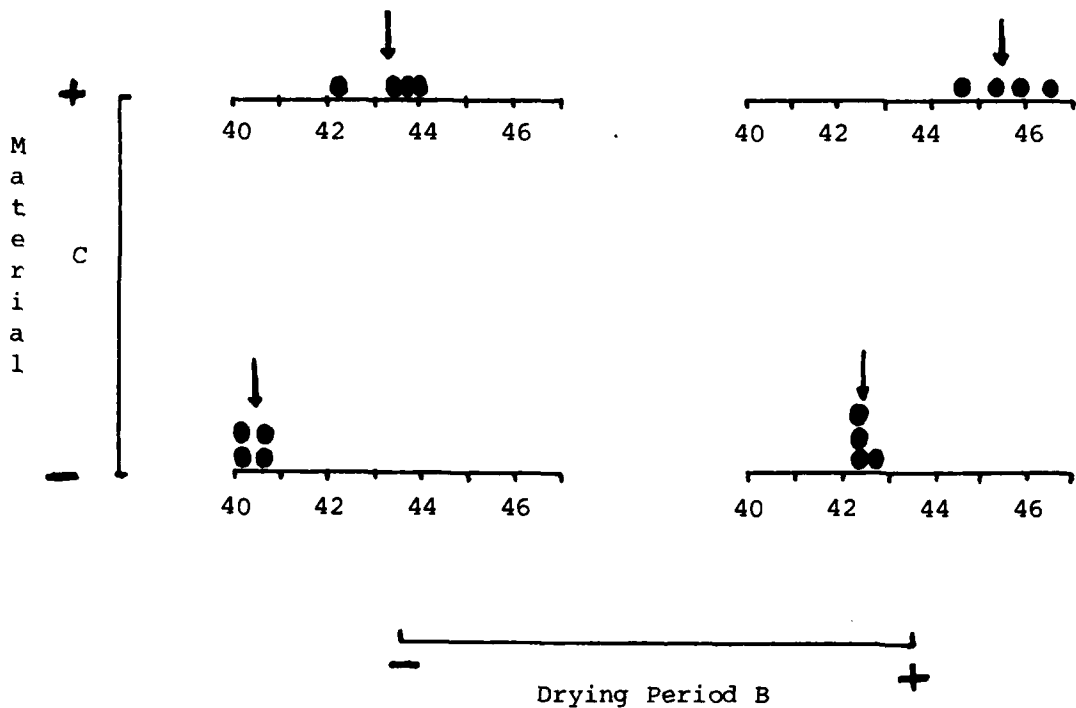


Figure 1. Tensile data as four replicates of a 2^2 factorial design in factors B and C only.

*Data of this kind might be accounted for by the effect of one or more variables other than B that affected tensile strength only at the "plus level" of C (only when the alternative material was used). Analysis of the eight runs made at the plus level of C does not support this possibility, however.

4. RATIONALES FOR USING SCREENING DESIGNS

Before proceeding we need to consider the question, "In what situations are screening designs, such as highly fractionated factorials, useful?"

4.1. Effect Sparsity

A common industrial problem is to find from a rather large number of factors those few that are responsible for large effects. The idea is comparable to that which motivates the use in quality control studies of the "Pareto diagram." (See, for example, Ishikawa (1976)). The situation is approximated by postulating that only a small proportion of effects will be "active" and the rest "inert". We call this the postulate of effect sparsity. For studying such situations, highly fractionated designs and other orthogonal arrays (Finney (1945), Plackett and Burman (1946), Rao (1947), Taguchi and Wu (1980)) which can screen moderately large numbers of variables in rather few runs are of great interest. Two main rationalizations have been suggested for the use of these designs; both ideas rely on the postulate of effect sparsity but in somewhat different ways.

4.2. Rationale Based on Prior Selection of Important Interactions

It is argued (see for example Davies (1954)) that in some circumstances physical knowledge of the process will make only a few interactions likely and that the remainder may be assumed negligible. For example, in the welding experiment described above there were 36 possible two-factor interactions between the nine factors, but only four were regarded as likely, leaving 32 such interactions assumed negligible. The difficulty with this idea is that in many applications the picking out of a few "likely" interactions is difficult if not impossible. Indeed the investigator might justifiably protest that, in the circumstance where an experiment is needed to determine which first order (main) effects are important, it is illogical that he be expected to guess in advance which effects of second order (interactions) are important.

4.3. Projective Rationale Factor Sparsity

A somewhat different notion is that of factor sparsity. Thus suppose that, of the k factors considered, only a small subset of unknown size d , whose identity is however unknown, will be active in providing main effects and interactions within that subset. Arguing as in Box and Hunter (1961) a two-level design enabling us to study such a system is a fraction of resolution $R = d + 1$ (or in the terminology of Rao (1947) an array of strength d) which produces complete factorials (possibly replicated) in every one of the $\binom{k}{d}$ spaces of $d = R - 1$ dimensions. For example, we have seen that on the assumption that only factors B and C are important, the welding design could be regarded as four replicates of a 2^2 factorial in just those two factors. But because the design is of resolution $R = 3$ the same would have been true for any of the 36 choices of two out of the nine factors tested. Thus the design would be appropriate if it were believed that not more than two of the factors were likely to be "active".

	Columns	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
(a)	2_{III}^{15-11}
(b)	2_{IV}^{8-4}
(c)	2_V^{5-1}
(d)	2^4

TABLE 3. Some alternative uses of the orthogonal array of Table 2.

For further illustration we consider again the sixteen-run orthogonal array of Table 2 and, adopting a roman subscript to denote the resolution R of the design, we indicate in Table 3 various ways in which that array might be used. It may be shown that

(a) If we associated the fifteen contrast columns of the design with fifteen factors, we would generate a 2_{III}^{15-11} design providing four-fold replication of 2^2 factorials in every one of the 105 two-dimensional projections.

(b) If we associated only columns 1, 2, 4, 7, 8, 11, 13, and 14 with eight factors we would generate a 2_{IV}^{8-4} design providing two-fold replication of 2^3 factorials in every one of the 56 three-dimensional projections.

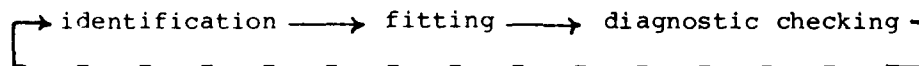
(c) If we associated only columns 1, 2, 4, 8, and 15 with five factors we would generate a 2_V^{5-1} design providing a 2^4 factorial in every one of the four-dimensional projections.

(d) If we associated only columns 1, 2, 4, and 8 with four factors we would obtain the complete 2^4 design from which this orthogonal array was in fact generated.

Designs (a), (b) and (c) would thus be appropriate for situations where we believed respectively that not more than 2, 3, or 4 factors would be active. Notice that intermediate values of k could be accommodated by suitably omitting certain columns. Thus the welding design is a 2_{III}^{9-5} arrangement which can be obtained by omitting 6 columns from the complete 2_{III}^{15-11} . Notice finally that for intermediate designs we can take advantage of both rationales by arranging, as was done for the welding design, that particular interactions are isolated.

* The designs give partial coverage for a larger number of factors, for example (Box and Hunter (1961)) 56 of the 70 four-dimensional projections of the 2_{IV}^{8-4} yield a full factorial in four variables.

A discussion of the interactive model building process by Box and Jenkins (1970) characterized three steps in the iterative data analysis cycle indicated below



Most of the present paper is concerned with model identification - the search for a model worthy to be formally entertained and fitted by an efficient procedure such as maximum likelihood. The situation we now address concerns the analysis of fractional designs such as the welding design in the above context when only a few of the factors are likely to have effects but these may include dispersion effects as well as location effects.

5. DISPERSION EFFECTS

We again use the design of Table 2 for illustration. There are 16 runs from which 16 quantities -- the average and 15 effect contrasts -- have been calculated. Now if we were also interested in possible dispersion effects we could also calculate 15 variance ratios. For example, in column 1 we can compute the sample variance s_{1-}^2 for those observations associated with a minus sign and compare it with the sample variance s_{1+}^2 for observations associated with a plus sign to provide the ratio $F_1 = s_{1-}^2/s_{1+}^2$. If this is done for the welding data we obtain values for $\ln F_1$ * given in Figure 2(a). It will be recalled that in the earlier analysis a large dispersion effect associated with factor C (column 15) was found, but in Figure 2(a) the effect for factor C is not especially extreme, instead the dispersion effect for factor D (column 1) stands out from all the rest. This misleading indication occurs because we have not so far taken account of the aliasing of location and dispersion effects. Since sixteen linearly independent location effects have already been calculated for the original data, calculated dispersion effects must be functions of these. Recently (Box and Meyer 1984a) a general theory of location-dispersion aliasing has been obtained for factorials and fractional factorials at two levels. For illustration, in this particular example it turns out that the following identity exists for the dispersion effect F_1 , that is the F ratio associated with factor D and hence for column 1 of the design.

$$F_1 = \frac{(\hat{2}-\hat{3})^2 + (\hat{4}-\hat{5})^2 + (\hat{6}-\hat{7})^2 + (\hat{8}-\hat{9})^2 + (\hat{10}-\hat{11})^2 + (\hat{12}-\hat{13})^2 + (\hat{14}-\hat{15})^2}{(\hat{2}+\hat{3})^2 + (\hat{4}+\hat{5})^2 + (\hat{6}+\hat{7})^2 + (\hat{8}+\hat{9})^2 + (\hat{10}+\hat{11})^2 + (\hat{12}+\hat{13})^2 + (\hat{14}+\hat{15})^2} \quad (1)$$

Now (see Table 2) $\hat{14} = \hat{B} = 2.15$ and $\hat{15} = \hat{C} = 3.10$ are the two largest location effects, standing out from all the others. The extreme value of F_1 associated with an apparent dispersion effect of factor D(1) is largely

* In this figure familiar normal theory significance levels are also shown. Obviously the necessary assumptions are not satisfied in this case, but these percentages provide a rough indication of magnitude.

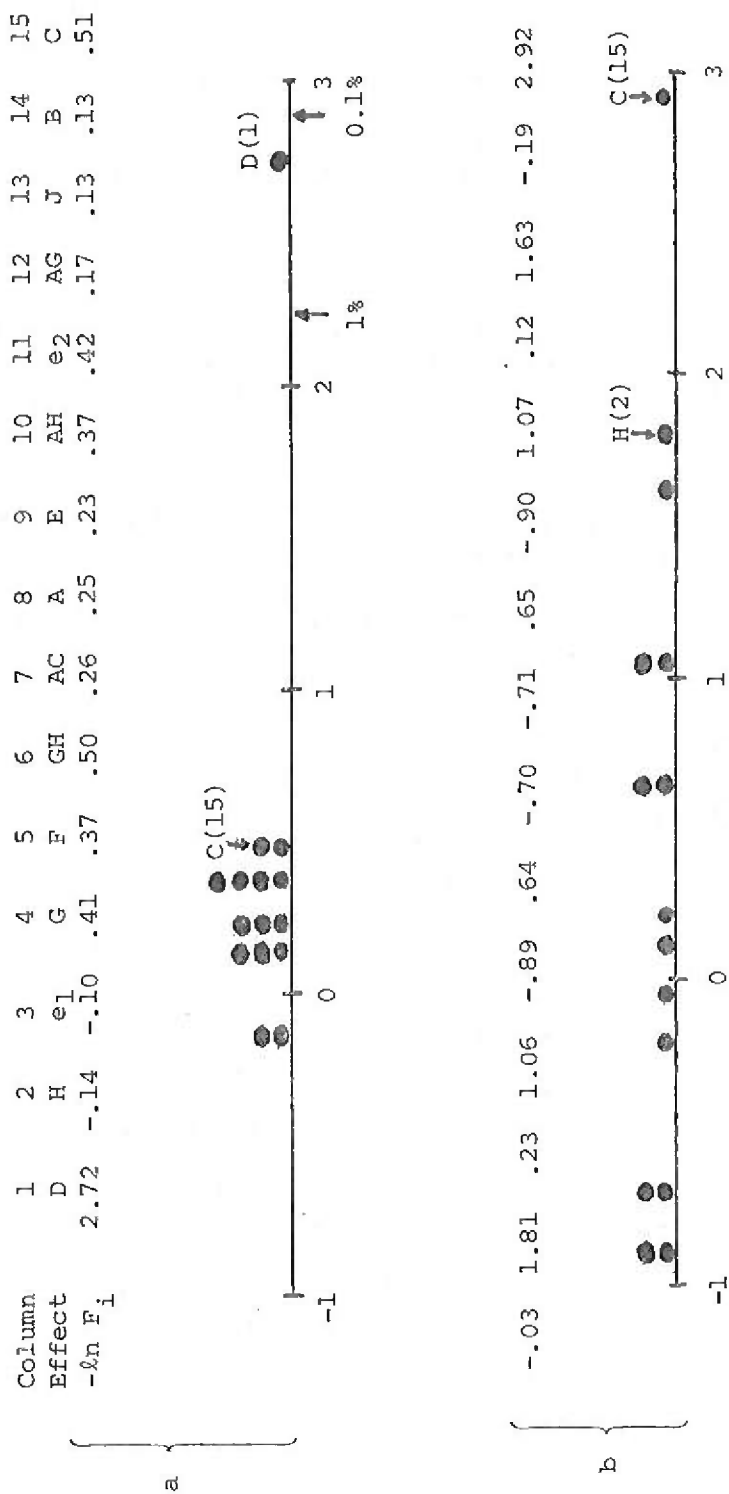


Figure 2. Welding experiment log dispersion effects (a) before, and (b) after elimination of location effects for B and C.

accounted for by the squared sum and squared difference of the location effects \hat{B} and \hat{C} which appear respectively as the last terms in the denominator and numerator of equation (1). A natural way to proceed is to compute variances from the residuals obtained after eliminating large location effects. After such elimination the alias relations of equation (1) remain the same except that location effects from eliminated variables drop out. That is zeros are substituted for eliminated variables. Variance analysis for the residuals after eliminating effects of B and C are shown in Figure 2(b). The dispersion effect associated with C (factor 15) is now correctly indicated as extreme. It is shown in the paper referenced above how, more generally, under circumstances of effect sparsity a location-dispersion model may be correctly identified when a few effects of both kinds are present.

6. ANALYSIS OF UNREPLICATED FRACTIONAL DESIGNS

Another important problem in the analysis of unreplicated fractional designs and other orthogonal arrays concerns the picking out of "active" factors. A serious difficulty is that with unreplicated fractional designs no simple estimate of the experimental error variance against which to judge the effects is available.

In one valuable procedure due to Cuthbert Daniel (1959, 1976) effects are plotted on Normal probability paper. For illustration Table 4 shows the calculated effects from a 2^{8-4}_{IV} design used in an experiment on injection molding (Box, Hunter and Hunter, 1978, p. 399). These effects are plotted on normal probability paper in Figure 3.

$T_1 = -0.7 \pm 1$	mold temp.
$T_2 = -0.1 \pm 2$	moisture content
$T_3 = 5.5 \pm 3$	holding pressure
$T_4 = -0.3 \pm 4$	cavity thickness
$T_5 = -3.8 \pm 5$	booster pressure
$T_6 = -0.1 \pm 6$	cycle time
$T_7 = 0.6 \pm 7$	gate size
$T_8 = 1.2 \pm 8$	screw speed
$T_9 = T_{1,2} = -0.6 \pm 1.2 + 3.7 + 4.8 + 5.6$	
$T_{10} = T_{1,3} = 0.9 \pm 1.3 + 2.7 + 4.6 + 5.8$	
$T_{11} = T_{1,4} = -0.4 \pm 1.4 + 2.8 + 3.6 + 5.7$	
$T_{12} = T_{1,5} = 4.6 \pm 1.5 + 2.6 + 3.8 + 4.7$	
$T_{13} = T_{1,6} = -0.3 \pm 1.6 + 2.5 + 3.4 + 7.8$	
$T_{14} = T_{1,7} = -0.2 \pm 1.7 + 2.3 + 6.8 + 4.5$	
$T_{15} = T_{1,8} = -0.6 \pm 1.8 + 2.4 + 3.5 + 6.7$	

TABLE 4. Calculated effects from a 2^{8-4}_{IV} design showing alias structure assuming three factor and higher order interactions negligible. Injection molding experiment.

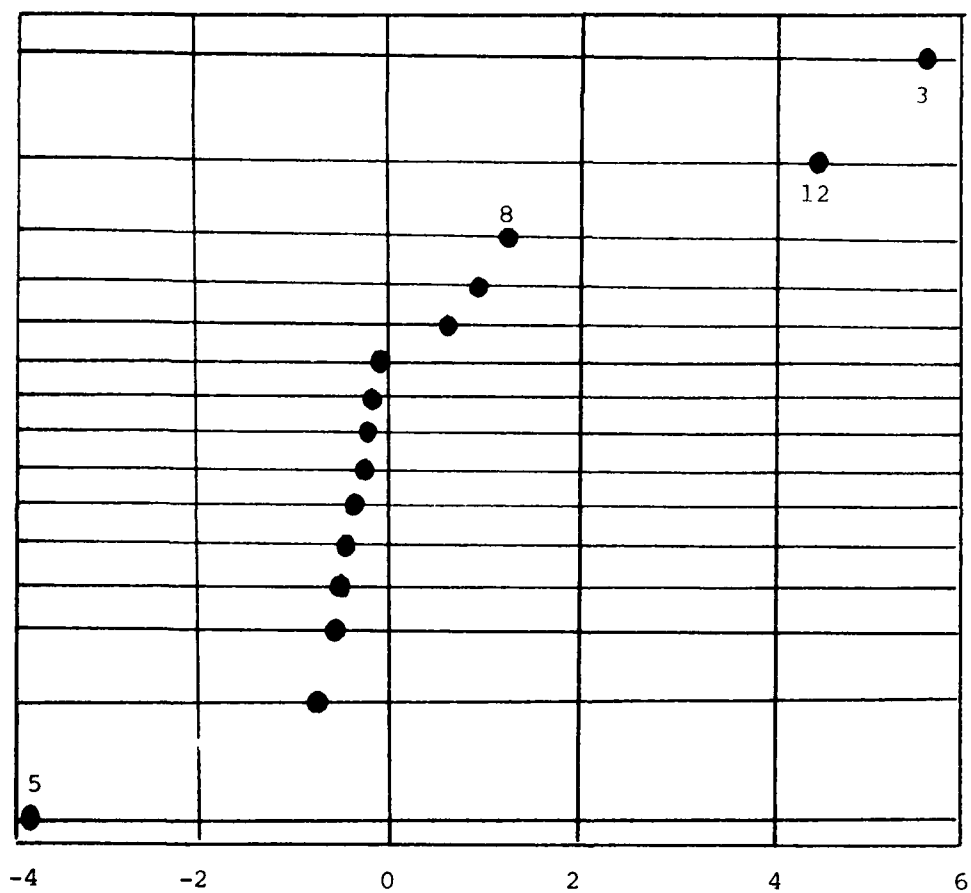


Figure 3. Normal plot of effects. Injection molding experiment.

An alternative Bayesian approach (Box and Meyer, 1984b) is as follows:
 Let T_1, T_2, \dots, T_v be standardized* effects with

$$T_i = e_i \quad \text{if effect inert}$$

$$T_i = e_i + \tau_i \quad \text{if effect active}$$

$$e_i \rightarrow N(0, \sigma^2), \quad \tau_i \rightarrow N(0, \sigma_\tau^2) \quad k^2 = \frac{\sigma^2 + \sigma_\tau^2}{\sigma^2}.$$

Suppose the probability that an effect is active is α .

Let $a_{(r)}$ be the event that a particular set of r of the v factors are active, and let $\tilde{T}_{(r)}$ be the vector of estimated effects corresponding to active factors of $a_{(r)}$. Then, (Box and Tiao, 1968) with $p(\sigma) \propto \frac{1}{\sigma}$ the posterior probability that $\tilde{T}_{(r)}$ are the only active effects is:

$$P[a_{(r)} | \tilde{T}, \alpha, k] \propto \left[\frac{\alpha k^{-1}}{1 - \alpha} \right]^r \left\{ 1 - \left(1 - \frac{1}{k^2} \right) \frac{S_{(r)}}{S} \right\}^{-\frac{v}{2}},$$

where $S_{(r)} = \tilde{T}_{(r)}' \tilde{T}_{(r)}$ and $S = \tilde{T}' \tilde{T}$. In particular the marginal probability that an effect i is active given T , α and k is proportional to

$$\sum_{\substack{a_{(r)} \\ i \text{ active}}} \left[\frac{\alpha k^{-1}}{1 - \alpha} \right]^r \left\{ 1 - \left(1 - \frac{1}{k^2} \right) \frac{S_{(r)}}{S} \right\}^{-\frac{v}{2}}.$$

A study of the fractional factorials appearing in Davies (1954), Daniel (1976) and Box, Hunter and Hunter (1978) suggested that α might range from 0.15-0.45 while k might range from 5 to 15. The posterior probabilities computed with the (roughly average) values. $\alpha = 0.30$ and $k = 10$ are shown in Figure 4(a) in which N denotes the probability (negligible for this example) that there are no active effects. The results from a sensitivity analysis in which α and k were altered to vary over the ranges mentioned above is shown in Figure 4(b).

It will be seen that Figure 4(a) points to the conclusion that active effects are associated with columns 3, 5 and 12 of the design and that column 8 might possibly also be associated with an active factor. Figure 4(b) suggests that this conclusion is very little affected by widely different choices for α and k . Further research with different choices of prior, with marginization with respect to k , and with different choices of the distribution assumptions is being conducted.

*For three-level and mixed two and three level designs for example, this analysis is carried out after the effects are scaled so that they all have equal variances.

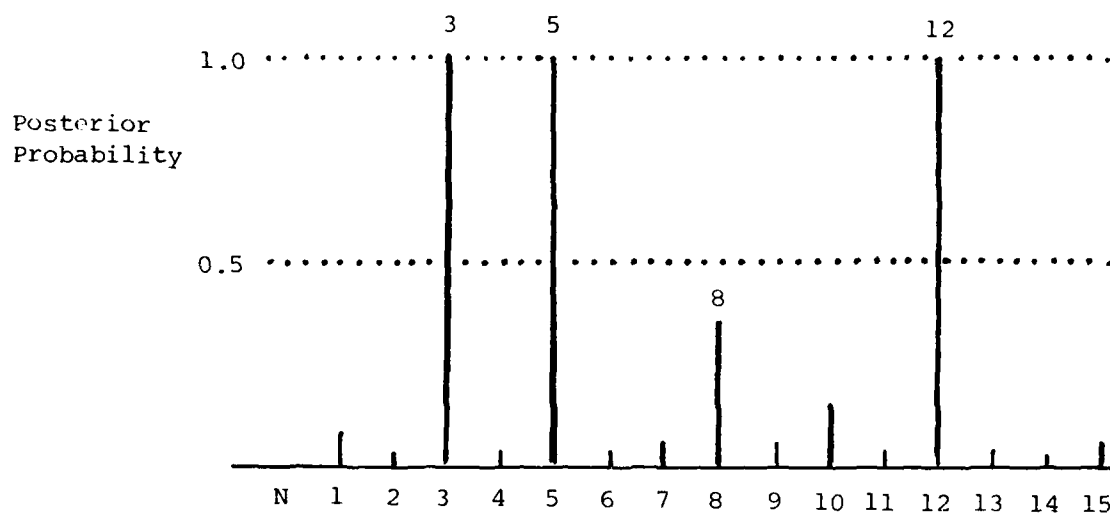


Figure 4(a) Welding experiment. Posterior probability that factor i is active ($\alpha = 0.30$, $k = 10$).

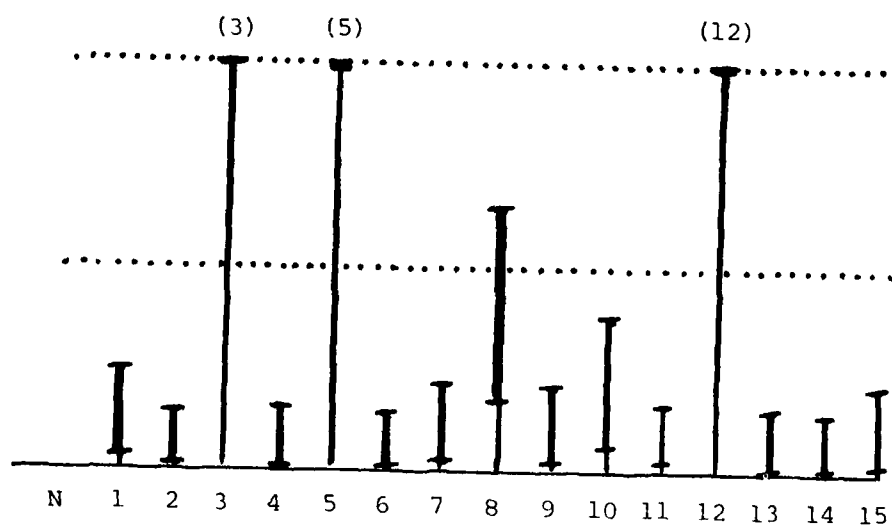


Figure 4(b) Sensitivity analysis for posterior probability $\alpha = .15 - .45$, $k = 5 - 15$.

7. OTHER RESEARCH

Topics which are emphasized in Taguchi's approach to "off line quality control" are (a) reduction of variation by error transmission studies and (b) the choosing of a product design so that it is robust with respect to environmental variables.

These topics are also receiving attention in further research.

REFERENCES

- Deming, W. Edwards (1982). Quality, Productivity and Competitive Position. M.I.T., Center for Advanced Engineering Study, Cambridge, MA 02139.
- Garvin, David A. (1983). Quality on the Line. Harvard Business Review, Vol. 61, No. 5, p. 64-75.
- Shewhart, W. A. (1931). The Economic Control of Quality of Manufactured Product. Van Nostrand (Reprinted in 1981 by American Society for Quality Control.)
- Taguchi, G. and Wu, Y. (1980). Introduction to Off-Line Quality Control. Central Japan Quality Control Association, Nagoya, Japan.
- Ishikawa, K. (1976). Guide to Quality Control. Asian Productivity Organization, Tokyo.
- Finney, D. J. (1945). The Fractional Replication of Factorial Arrangements. Annals of Eugenics, 12, 4, 291-301.
- Plackett, R. L. and Burman, J. P. (1946). Design of Optimal Multifactorial Experiments. Biometrika, 23, 305-325.
- Rao, C. R. (1947). Factorial Experiments Derivable from Combinatorial Arrangements of Arrays. J. Roy. Statist. Soc., B9, 128-140.
- Davies, O. L. Editor (1954). The Design and Analysis of Industrial Experiments. London: Oliver and Boyd.
- Box, G. E. P. and Hunter, J. S. (1961). The 2^{k-p} Fractional Factorial Designs. Technometrics, 3, 311, 449.
- Box, G. E. P. and Jenkins, G. M. (1970). Time Series Analysis, Forecasting and Control. San Francisco: Holden-Day.
- Box, G. E. P. and Meyer, R. D. (1984a). Analyzing Two-Level Fractional Factorial Designs for Possible Dispersion Effects. Mathematics Resesarch Center Technical Summary Report, Madison, WI 53705.
- Daniel, C. (1959). Use of Half-Normal Plots in Interpreting Factorial Two-Level Experiments. Technometrics, 1, 4, 149.

Daniel, C. (1976). Applications of Statistics to Industrial Experimentation. New York: Wiley.

Box, G. E. P., Hunter, W. G. and Hunter, J. S. (1978). Statistics for Experimenters. New York: Wiley.

Box, G. E. P. and Meyer, R. D. (1984b). An Alternative Approach for the Analysis of Fractional Designs. Mathematics Research Center Technical Summary Report, Madison, WI 53705.

Box, G. E. P. and Tiao, G. C. (1968). A Bayesian Approach to Some Outlier Problems. Biometrika, 55, 119.

Models of Military Combat with
Logistics Components

John S. Maybee*
Department of Mathematics
Campus Box 426
University of Colorado
Boulder, CO 80309

*Work supported in part by the U.S. Army under grant ARO #DAAG
29-80-K-0060 .

1. Combined arms models of combat. The classical Lanchester model of military combat is defined by the equations

$$\begin{aligned}\dot{x} &= -ay, \\ \dot{y} &= -bx,\end{aligned}$$

where $x(t)$ is the strength of the X-force at time t and $y(t)$ is the strength of the Y-force at time t . If we are interested in adjoining logistics considerations to combat models it is more realistic to start with combined arms models of combat.

A general combined arms model of Lanchester type can be formulated in the following way. The X-force is assumed to have m units of strengths $x_1(t), \dots, x_m(t)$ at time t and the Y-force has n units of strengths $y_1(t), \dots, y_n(t)$ at time t . These units may be of different types. Let $x = (x_1, \dots, x_m)$, $y = (y_1, \dots, y_n)$. The combat between X and Y forces is governed by the equations

$$(1) \quad \begin{cases} \dot{x} = -Ay, \\ \dot{y} = -Bx, \end{cases}$$

where A is an $m \times n$ matrix and B an $n \times m$ matrix. We also have $A \geq 0$ and $B \geq 0$, i.e. A and B are nonnegative matrices. The properties of the solution of the system (1) depend entirely upon the matrix

$$M = \begin{bmatrix} 0 & -A \\ -B & 0 \end{bmatrix},$$

an $(n+m) \times (n+m)$ matrix, subject to initial conditions

$$x(0) = x_0 \equiv (x_{10}, \dots, x_{m0}) ,$$

$$y(0) = y_0 \equiv (y_{10}, \dots, y_{n0}) .$$

We have $x_0 > 0$, $y_0 > 0$.

In such a model the elements of the matrixes A and B have the following forms: $a_{ij} = \lambda_{ij}\alpha_{ij}$ and $b_{ij} = \mu_{ij}\beta_{ij}$. The number α_{ij} represents the efficiency of the unit y_j of the Y -force when used against the unit x_i of the X -force . On the other hand, the number λ_{ij} represents the fraction of the firepower of unit y_i directed against the unit x_i by the Y -commander. We may suppose that $\sum_{i=1}^m \lambda_{ij} = 1$. The numbers β_{ij} and μ_{ij} are similarly defined. The numbers α_{ij} and β_{ij} are analogous to the coefficients a and b in the classical Lanchester model--thus we may call them attrition rate coefficients. The numbers λ_{ij} and μ_{ij} (note that $\sum_{i=1}^n \mu_{ij} = 1$ for each j) represent a priori choices which must be made by the commanders of the Y -force and the X -force, respectively.

An element a_{ij} of the matrix A can be zero if either the unit y_j is ineffective against the unit x_i or if the Y -commander elects not to use unit y_j against unit x_i . A similar meaning is attached to an element b_{ij} of B being zero.

An example of a combined arms model is the following which includes X and Y units of four types:

- (i) Direct fire combat unit.
- (ii) Artillery unit;
 - (a) direct support,
 - (b) counter battery fire,
 - (c) air defense suppression.

- (iii) Air support unit;
 - (α) close support,
 - (β) air defense suppression,
 - (γ) artillery suppression.
- (iv) Air defense unit.

Here we have listed the functions each unit can perform. The model then takes the form

$$M_1 = \begin{bmatrix} 0 & -A_1 \\ -B_1 & 0 \end{bmatrix}$$

where

$$A_1 = \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 \\ 0 & a_{22} & a_{23} & 0 \\ 0 & 0 & 0 & a_{34} \\ 0 & a_{42} & a_{43} & 0 \end{bmatrix}$$

and B_1 has the same form as A_1 . The matrix M_1 is reducible so the model can be subdivided into two submodels. The main submodel has matrix

$$M = \begin{bmatrix} 0 & 0 & 0 & -a_{22} & -a_{23} & 0 \\ 0 & 0 & 0 & 0 & 0 & -a_{34} \\ 0 & 0 & 0 & -a_{42} & -a_{43} & 0 \\ -b_{22} & -b_{23} & 0 & 0 & 0 & 0 \\ 0 & 0 & -b_{34} & 0 & 0 & 0 \\ -b_{42} & -b_{43} & 0 & 0 & 0 & 0 \end{bmatrix}.$$

A satisfactory theory can be developed if we assume M is a Lanchester Matrix (see Maybee 1984). This means A and B are regular, i.e. each row and column contains at least one nonzero element, and M is irreducible. In the above example the matrix M is a Lanchester Matrix.

2. Supply models. For the simplest supply models we adjoin to the system (2) supply vectors $Sx(t) = (Sx_1, \dots, Sx_m)$, and $Sy(t) = (Sy_1, \dots, Sy_n)$. Here $Sx_i(t)$ is the supply level of unit x_i of X-force at time t . The function $Sy_i(t)$ has a similar meaning.

The equations of combat are supplemented by

$$(2) \quad \dot{Sx} = -D_1 x, \quad \dot{Sy} = -D_2 y$$

where D_1 and D_2 are diagonal matrices of size $m \times m$ and $n \times n$, respectively, with positive diagonal elements. Thus the supply levels of the units diminish with time at a rate proportional to the unit level.

The system of equations (1), (2) is to be solved subject to the following stopping rules. If a unit level or supply level falls below an acceptable percentage of initial unit or initial supply level, then that unit is withdrawn from combat.

Note that in this model, once the equations (1) are solved, equations (2) can be explicitly integrated to determine the vectors $Sx(t)$ and $Sy(t)$. Thus the theory of system (1) is immediately applicable to it.

A more sophisticated supply model can be developed as follows. Define the vectors $u(t) = (u_1(t), \dots, u_p(t))$ and $v(t) = (v_1(t), \dots, v_q(t))$ which represent supply lines for the X-force and Y-force, respectively. Here $u_i(t)$ is the capacity of the i -th X-force supply line at time t . The function $v_i(t)$ has a similar meaning.

In this version of the model we assume supplies are brought in as combat goes on. Also each commander directs fire against opposing supply lines. Thus to the equations (1) we now adjoin the equations

$$(2') \quad \begin{cases} \dot{S}_x = -D_1 x + \tilde{A}u , \\ \dot{S}_y = -D_2 y + \tilde{B}v \end{cases}$$

and

$$(3) \quad \begin{cases} \dot{u} = -Ey , \\ \dot{v} = -Fx . \end{cases}$$

Here \tilde{A} is an $m \times p$ matrix, \tilde{B} is $n \times q$ matrix, E is a $p \times n$ matrix, and F is a $q \times m$ matrix. All of these matrices are nonnegative. Initial values are given for S_x , S_y , u and v where $S_x(0) > 0$, $S_y(0) > 0$, $u(0) > 0$, $v(0) > 0$. The elements of the matrices \tilde{A} , \tilde{B} , E , and F have the same form as those in the matrices A and B . This is because each commander makes a priori decisions as to how he uses his supply lines to bring in supplies and also how much of his firepower he directs against each of his opponents' supply lines.

Note that in this model it is still true that the solution of the system (1) completely determines the entire model. Once the system (1) has been solved the system (3) can be integrated directly to determine the vectors u and v . Then the system (2') can be integrated to determine the vectors S_x and S_y .

Nevertheless we have a significant new issue introduced here. This is because of the question of how much firepower should be used against supply lines versus how much is used against opposing combat units. This version of

the model permits us to evaluate the effects of such choices upon the outcome of combat. The previous simpler version only allows us to evaluate the effects of supply levels upon the outcome of combat.

A very sophisticated supply model can now be formulated. We now assume that the supply lines are also used to reinforce the combat units. We then have an entirely new model having the form

$$(4) \quad \left\{ \begin{array}{l} \dot{x} = -A_1 y + A_2 u, \quad \dot{u} = -E y, \\ \dot{y} = -B_1 x + B_2 v, \quad \dot{v} = -F x, \\ \dot{S}_x = -D_1 x + \tilde{A} u, \\ \dot{S}_y = -D_2 y + \tilde{B} v. \end{array} \right.$$

Here A_2 is an $m \times p$ matrix and B_2 is an $n \times q$ matrix, $A_2 \geq 0$, $B_2 \geq 0$. The elements of these matrices have the same form as the matrices A_1 and B_1 , that is, they are the products of coefficients which define the capability of each supply line to furnish the given type of reinforcements (men, tanks, etc.) with a number on the interval $[0,1]$ which represents the fraction of supply line capacity devoted to such reinforcements.

The matrix of the system (4) is

$$N = \begin{bmatrix} 0 & A_2 & -A_1 & 0 \\ 0 & 0 & -E & 0 \\ -B_1 & 0 & 0 & B_2 \\ -F & 0 & 0 & 0 \end{bmatrix}$$

and the system to be solved is

$$(5) \quad \begin{bmatrix} \dot{x} \\ \dot{u} \\ \dot{y} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & A_2 & -A_1 & 0 \\ 0 & 0 & -E & 0 \\ -B_1 & 0 & 0 & B_2 \\ -F & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ u \\ y \\ v \end{bmatrix}$$

We call N a Lanchester logistics matrix if each of the matrices A_1, B_1, A_2, B_2, E, F is regular and N is irreducible. Of course, systems for which N is reducible can be decomposed into smaller irreducible systems. Also once (5) has been solved, we may again determine Sx and Sy by integration.

Preliminary results show that the basic properties of the solution of (5) depend only upon the structure of the matrix N . Thus all such logistics systems can be expected to have similar solutions.

3. Issues that can be addressed by such models. It is important to understand first how our models should be used. Because of the fact that a large set of a priori decisions must be made by each commander with regard to how he allocates his firepower and how he uses his supply lines, it is reasonable to suppose that a given matrix M or N will apply from time 0 to the first time t_1 at which one of the commanders changes his allocations. Then M is replaced by a new matrix \tilde{M} (N by \tilde{N}) and a model of the same general form holds until time t_2 . Of course, the initial values for the interval $[t_1, t_2]$ are the same as the final values at t_1 using the matrix M or N . Thus a lengthy combat can be modeled as a sequence of such models. We can even use our models for combat which lasts over a period of days or weeks with intermittent periods of quiet (say at night) during which supplies or reinforcements are brought in. Then the initial conditions for $[t_1, t_2]$ would not necessarily be the same as the final conditions on $[0, t_1]$.

A variety of problems may be solved using these models. We may investigate the effect of a priori decisions made by the commanders upon the battle progress. To do this it is necessary to devise various measures of combat effectiveness (see Willis 1982 for a variety of such measures applied to classical combat models). We can study the result of invoking various stopping rules and the effect of initial force sizes on battle progress. All of these issues can be studied using combined arms models.

The issues mentioned above are also relevant to the various supply models. But we may use the supply models to try and understand the answers to other questions. For example, what is the relation between combat levels and supply levels and, in particular, what are the optimal initial supply levels? How should supply line capacities be allocated so as to insure against having to withdraw from combat because of inadequate supplies? Conversely, what is the most effective allocation of fire power between enemy units and enemy supply lines? Deeper issues concern questions such as when a commander should change his allocations and how.

References

1. Maybee, J.S., The theory of combined arms Lanchester-type models of warfare, U.S. Naval Log. Quart. (in press).
2. Willis, R.F., Models, measures of effectiveness and scenario assumptions --a sensitivity analysis. The Royal Military College of Science, Dept. of Management Sciences working paper OR/WP/19 (1982), (Shrivenham, U.K.)

OVERVIEW OF EXPERT SYSTEMS

Capt Stephen E. Cross, USAF
Artificial Intelligence Laboratory
Air Force Institute of Technology
Wright-Patterson AFB, Ohio 45433

Introduction

An expert system is a computer program that embodies the knowledge of a human expert and a reasoning process (which may be suggestive of the human expert's reasoning process) to perform some problem solving task that is usually deferred to a human expert. Often, these programs are referred to as knowledge based programs or intelligent assistants. Many programs can be considered to perform some expert task. For instance, a FORTRAN program that computes a Fourier transform accomplishes something that would be difficult for a human to do. What distinguishes expert systems in the artificial intelligence context, is that these programs use the same type of experiential knowledge as do their human counterparts. A significant architectural characteristic is that this knowledge is contained in a separate knowledge base.

Expert systems can perform many tasks. A taxonomy of tasks include: prediction, diagnosis, design, planning, monitoring, interpretation, debugging, repair, instruction, and control.

The type of knowledge that human experts use can be divided into three categories: facts, heuristics, and beliefs. Facts are perhaps the easiest form of knowledge to visualize. They are just static pieces of data which are thought to be true. For instance, a fact is 'the wing span of a T-38 is 24 feet.' Heuristics are pieces of experiential knowledge which are most often stated in the form of production rules. Heuristics are 'rules of thumb' or gut feelings that are acquired throughout the course of a career. They are rarely recorded in text books or professional articles. An example heuristic is taken from MYCIN [ref 1] an expert system that performs the diagnosis and recommended treatment task in infectious blood disease.

If 1) the infection is primary-bacteremia,
 2) the site of the culture is one of the sterile sites,
 3) the suspected portal of entry of the organism is the
 gastrointestinal tract,

Then there is suggestive evidence (cf=.7) that the identity
 of the organism is bacteroides.

The computer representation of this rule would look like:

```
(IF (AND (SAME CNTXT INFECT PRIMARY-BACTEREMIA)
         (MEMBF CNTXT SITE STERILESITES)
         (SAME CNTXT PORTAL GI)))
```

(TH (CONCLUDE CNTXT IDENT BACTEROIDES TALLY .7)

An expert system requires knowledge of belief. Belief enables the computer to decide how much credibility to attach to facts or heuristics. Quite often belief has been represented probabilistically, but symbolic representations of belief are now becoming popular. MYCIN's use of certainty factors is typical of numerical belief representation. Belief is mapped into the range $[-1,1]$ where 1 represents being certain something is true, -1 represents being certain something is not true, and 0 represents the lack of any knowledge to believe or not believe something. The above rule has a cf of .7 which indicates a fairly certain level of belief (equivalent to a physician saying 'I'm fairly certain'). An algorithm is used to combine cf's during a search process for applicable rules so that the path with the highest combined cf is evaluated first.

An Example.

As a prototypical expert system, I will discuss the animal production system of Winston and Horn [ref 2]. This system is used to identify animals in a zoo. Although simple, it illustrates how more complicated systems like MYCIN operate. An understanding of production system operation is a prerequisite for understanding more complicated rule-based system architectures.

A production system consists of a rule-base, a data base, and a control program. The rule base is the repository of all heuristics. In theory, the rule base is unordered. That is, there is no significance in contiguous rules. Some systems (e.g., MYCIN) include certainty factors which are processed to give a measure of belief. In the animal system there is no uncertainty, hence there is no need for certainty factors. The data base in MYCIN consists of facts gleaned from the patient history (e.g., the subject smokes 3 packs a day) and results of laboratory tests. In the animal system the data base is a list of symbolic facts. The list is a repository for known characteristics of animals. The control structure uses backward chaining. For instance, if the computer wanted to deduce that the patient had a particular disease, it would obtain a list of rules that made conclusions about that disease. Then the antecedents of each such rule would be tested. Antecedents for which there is insufficient data would be defined as subgoals and rules that made conclusions about these new subgoals would be accessed, hence backward chaining.

Many other control schemes are possible. The process of backward chaining is often called goal-directed search. Another process, forward chaining, seeks to invoke rules whose antecedents presently match the data base. This strategy is called data-driven search. Combinations of backward and forward chaining are often employed in production systems. McDermott [ref 3] has proposed several variations. Rather than a pure forward or backward search, McDermott suggests keeping track of rules that have been applied successfully in the past and trying them first,

or trying rules first that have what are judged to be computationally tractable antecedents.

The animal production system consists of lists of facts and rules. The following trace will illustrate some of the basic concepts.

```
; the data base initially consists of these known facts
(setq facts '((animal has hair)
              (animal eats meat)
              (animal has tawny color)
              (animal has dark spots)))

;the rule base consists of rules like
(setq rules '((rule id1 (if (animal has hair)
                           (then (animal is mammal)))
              (rule id2 (if (animal gives milk)
                           (then (animal is mammal)))
              (rule id3 (if (animal has feathers)
                           (then (animal is bird)))
              (rule id4 (if (animal flies)
                           (animal lays eggs)
                           (then (animal is bird)))
              (rule id5 (if (animal eats meat)
                           (then (animal is carnivore)))
              (rule id6 (if (animal has pointed teeth)
                           (animal has claws)
                           (animal has forward eyes)
                           (then (animal is carnivore)))
              (rule id7 (if (animal is mammal)
                           (animal has hoofs)
                           (then (animal is ungulate)))
              (rule id8 (if (animal is mammal)
                           (animal chew cud)
                           (then (animal is ungulate)
                                (animal is even toed)))
              (rule id9 (if (animal is carnivore)
                           (animal has tawny color)
                           (animal has black stripes)
                           (then (animal is cheetah)))
              .
              .
              .
```

The animal production system can be run in the forward or backward chaining mode. In the forward mode, a search will be made for rules whose left side match the data stored in the facts list. Any rule that matches (and whose right side has not been previously written into the list) will have its right side added to the list. The search continues until no more rules are applicable. In this system the function deduce accomplishes this search.

```
(deduce)
rule id1 deduces (animal is mammal)
```



```
rule id5 deduces (animal is carnivore)
rule id9 deduces (animal is cheetah)
```

At this point the facts list becomes:

```
((animal has hair) (animal eats meat)
(animal has tawny color) (animal has dark spots)
(animal is mammal) (animal is carnivore)
(animal is cheetah))
```

The system can be run in a backward chaining mode by establishing a list of hypotheses.

```
(setq hypotheses '((animal is cheetah)
                   (animal is ostrich)
                   (animal is penguin)
                   (animal is cow)
                   (animal is elephant)
                   .
                   .
                   .)
```

A search is made for rules whose left side support each hypothesis. The function that performs this task is `diagnose`.

```
(diagnose)
Is it true: (animal has feathers)? no    ;the user responds
Is it true: (animal flies)? no
Is it true: (animal has hair)? yes
rule id1 deduces (animal is mammal)
Care to know how? yes
(animal is mammal) demonstrated by: (animal has hair)
Is it true: (animal has hoofs)? why      ;the user can
;ask why such a question was asked
(animal has hoofs) needed to show (animal is ungulate)
Is it true: (animal has hoofs)? yes
rule id7 deduces (animal is ungulate)
Care to know how? no
Is it true: (animal has black stripes)? yes
rule id12 deduces (animal is zebra)
Care to know how? no
Hypothesis (animal is zebra) is true.
any other questions? no
```

The production system approach has several advantages. The knowledge base is separate from the control program, hence it is easy to modify, add, or delete knowledge. Modification of the knowledge base does not inhibit operation of the computer program. Knowledge has a uniform representation.

A major advantage of the production system approach is that the program can 'explain' its solutions by reciting some portion of the rules that were used in the reasoning process. For example, a backward chaining production system interprets questions like 'How?' to mean 'How did I reach this conclusion' and

will list the rules that were instrumental in the decision. Questions like 'Why?' (why was this question asked) are answered by listing the antecedents of the rule to which the questions context is the subgoal. Sample questions and answers are shown in the MYCIN trace above. It should be noted that this is a very primitive form of computer understanding. A truly intelligent explanation facility would have powerful 'truth maintenance' facilities and access to first principles. An interesting case study was conducted with MYCIN about four years ago. It was decided that since MYCIN was so splendid at diagnosis and had an explanation capability, that it would make a good medical instructor. But it was soon found that an experiential knowledge base was deficient at explaining many of the causal relationships in medicine. For instance, MYCIN has a rule that says 'don't administer tetracycline to children under eight years of age.' No where in the knowledge are the facts that tetracycline inhibits bone development, a physiological piece of knowledge.

There are several problems with production systems at the present. As stated above, an expert system needs access to large stores of knowledge. Some of the knowledge is experiential and is probably best represented as production rules. Other knowledge concerns domain theories. CASENET [ref 4] represents causal relationships in internal medicine. MDX [ref 5] orders disease processes in a tree structure and records at each node only that knowledge that is required to establish the existence of that disease process. Cross' air traffic control system [ref 6] uses a network representation for control algorithms. New heuristics are justified by propagating values (the effect of applying heuristics) throughout the network.

MYCIN led to the development of many expert systems. Since the knowledge base was separate from the program, it is possible in many applications to simply insert a new knowledge base for a different domain. PUFF [ref 7] an expert system for the diagnosis of pulmonary lung disease, was written in EMYCIN (essentially MYCIN). PUFF demonstrates that a new expert system can be built in a fairly short period of time provided: 1) the domain of application is not sufficiently changed, and 2) the people building the expert system have experience in building knowledge based systems. PUFF was created in about 100 man hours by a team of expert knowledge engineers and physicians.

One major bottleneck in expert system design is knowledge acquisition. The speed at which an expert system can be built is directly related to the skill and experience of the expert system builder, commonly called the knowledge engineer. It is his job to become sufficiently conversant in the domain to talk intelligently with a cooperative domain expert, and to obtain the heuristics that the domain expert uses to do difficult problem solving. It's a paradox that domain experts have only vague ideas of the actual heuristics that they use. This is why training programs like medicine, law, etc. take many years. Michie [ref 8] relates an interesting example. A cheese manufacturing company in England relied on the skill of an elderly gentleman to do quality control

of its products. He could assess the quality of cheese by probing his finger through the wax seal and feeling the cheese inside. Because of his prowess and the fact that he was quite elderly, the company management wanted to automate his expert behavior. They brought in many mechanical engineers to attempt to build a device which could probe the cheese in the same manner as the expert. What the expert did not realize and hence was unable to verbalize to the system builders, was that his probing was simply a mental device he employed to focus his sense of smell on the particular cheese in front of him.

Another problem with rule bases is that they become large and search becomes computationally expensive. TIERESIAS, the subject of Randall Davis' PhD work [ref 9], used meta-rules to guide the invocation of domain rules. A meta-rule is simply a production rule that makes conclusions about domain rules. An example which was used in conjunction with a MYCIN-like system for investment advice is shown below:

If 1) the age of the investor is greater than 50,
 2) the investor is not independently wealthy,
Then there is evidence (cf=1.0) that only stocks that
 have high dividends should be considered.

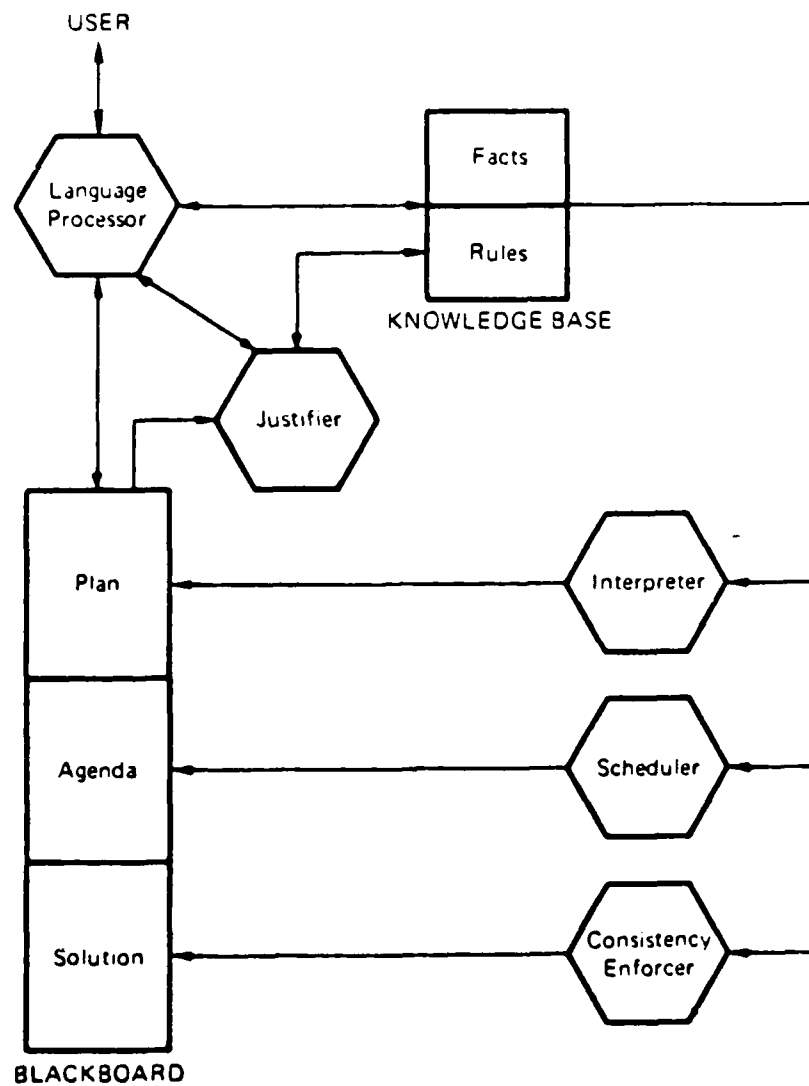
Another approach to knowledge base organization was offered by Aikins [ref 10] in her PhD dissertation. She noted the search inefficiency problems in PUFF and developed a frame based expert system where rules were organized into disease groups.

The final problem in rule bases which will be discussed here is belief justification. It is very important that only one domain expert be consulted in the creation of a new knowledge base. Often one expert's heuristic will contradict another expert's. The computer at present has no mechanism for truth maintenance [ref 11] although research into this area is proceeding.

An Ideal Architecture.

One should remember that a production system is only a simple architecture of an expert system. The anatomy of an 'ideal' expert system is shown in Fig. 1 [ref 12:17]. The system consists of a natural language front-end to facilitate communication between the computer and the user. Brooke's thesis [ref 13] describes a universal natural language front-end for expert systems. The blackboard is used to record intermediate results posted for use by many knowledge bases which may be operating in parallel. The knowledge base contains facts as well as heuristic problem-solving rules. The interpreter controls how the knowledge base is searched. The enforcer adjusts previous conclusions when new data or knowledge alter their bases of support. The justifier rationalizes and explains the system's behavior.

Hearsay II [ref 14], a system designed to do speech recognition, embodies many of the concepts from the ideal architecture. Hearsay II was organized as a body of cooperating, independent



Ideal Architecture
Figure 1

specialists. Each specialist used knowledge that was unique to the speech recognition task. For instance, there was a knowledge base of rules for inferring phonemes and a separate knowledge base of grammatical rules.

Many other architectures exist. OPS5 [ref 15] allows control rules to be specified which allow different search strategies to be invoked in different contexts. R1 [ref 16] is a VAX computer configuration expert system written in OPS5. ROSS [ref 17] facilitates communicate between small experts (or actors) through a message passing language. ROSIE [ref 18] provides a structured English interface to facilitate representation of rules. All of these systems are variations on the production system theme.

Expert System Advice.

Penny Nii (an expert expert system builder) has offered some practical advice for those wishing to build an expert system [ref 19].

1. Don't be your own expert. It is hard to be objective about your own 'expert' knowledge.

2. The problem must be well chosen. AI is not the answer to every problem. Expert systems work best when the problem is well bounded. This means that while we can represent large amounts of problem specific knowledge we do not have a good handle on representing general world knowledge.

3. You need to meet the human expert more than halfway. Nii begins a new expert system building task by reading all the literature in the application domain.

4. If none of the tools that you have available will work, build one.

5. One needs a way to handle uncertainty. A weighting process must be built in that handles facts or knowledge like 'I strongly believe' or 'It might cause

6. The program must have easy means of knowledge base modification. The program must be able to explain its answers. Both imply that if the expert is to be a personal assistant to a human, that it should have a useful natural language front end.

Limitations.

There are some serious limitations to expert systems at the present time. Expert system techniques have to date only been successful in domains where the experiential knowledge of the expert could be decoupled from the world and common-sense knowledge of the expert. These programs tend to be idiot savants in that they neither recognize an interesting problem or solution and degrade quickly near the fringes of their knowledge. For instance, MYCIN fails when multiple diseases are present in the

body causing some infectious blood disease symptoms to be masked. Another limitation is that to date, expert systems have only been successfully applied to domains that are very narrow. The computer does a fantastic job at 'deep' inferencing. We do not have the capabilities to represent 'broad' knowledge in useful ways. Expert systems do not have the capability to do common sense reasoning. For instance, an expert system for the flight domain might represent emergency checklist 'scripts'. One of these states that when the cabin depressurizes, descend. However, an intelligent being would immediately rule out a descent when flying in the mountains.

All Is Not Lost.

Even with the limitations, there are many successful applications. Measures of success are 1) the number of companies that are building expert systems for internal use (e.g., Westinghouse, General Electric, NCR), 2) the amount of venture capital available to build systems for stock market analysis, etc., and 3) the huge salaries available to knowledge engineers (up to \$70,000 on the west coast). We conclude the section on expert systems with a listing of the known expert systems. Much of the list is taken from [ref 19] and is supplemented with systems that we have worked on.

Air Force Institute of Technology

- ATC (an air traffic control expert system framework)
- Maintenance expert systems (battle damage assessment, circuit card diagnosis, tech order automation)
- Military planning (several military planning aids)
- Natural language front ends to expert systems
- Pilot aids, ongoing research in advanced expert system architectures
- SPEREXAS (a speech recognition system)

Bioengineering

- MOLGEN (genetic experiment planning aid)

Chemistry

- DENDRAL (interprets mass spectrometer data)
- DECS (organic synthesis planning)

Computer Systems

- DART (diagnosis of computer faults)
- R1 (configure VAX systems)
- SPEAR (analysis of computer error logs)
- XSEL (assists sales people in selecting appropriate computer systems)

Engineering

- SACON (aids structural engineers)

General Purpose

- AGE (guides development of expert systems involving hypothesis formation and information fusion)
- AL/X (assists diagnostic experts)
- EMYCIN (MYCIN without the knowledge base)
- EXPERT (an inference system used in oil exploration tasks)
- KAS (an experimental knowledge acquisition system)
- LOOPS (an experimental knowledge representation system)
- OPS (a basic inference system)
- ROSIE (a basic inference system)
- TIERESIAS (aids in knowledge acquisition)
- UNITS (an early version of LOOPS)

Law

- LDS (an experimental system that models legal decision making)
- TAXMAN (an experimental system that deals with rules implicit in tax laws)

Maintenance

- CAT-1 (diagnosis of diesel train engines)

Military

- AIRPLAN (an expert system for air traffic control around aircraft carriers)
- HASP (an expert system for identification and tracking of ships using ocean sonar signals)
- KNOBS (a tactical aircraft planning aid)
- TATR (an expert system for tactical air targeteering, uses ROSIE)
- SWIRL (a tactical aircraft planning aid, uses ROSS)

Resource Exploration

- DIPMETER ADVISOR (analyzes information from oil wells)
- DRILLING ADVISOR (diagnosing oil well drilling problems)
- PROSPECTOR (evaluates sites for potential mineral deposits)

Medicine

- CADUCEUS (differential diagnosis in internal medicine)
- CASNET (a causal network that associates treatments with various diagnostic hypotheses)
- MYCIN (diagnoses infectious blood diseases)
- MDX (uses compiled knowledge to performed various diagnosis tasks)
- ONCOCIN (a management system for cancer chemotherapy)
- PUFF (diagnosis of pulmonary disorders, uses EMYCIN)

References

1. Shortliffe, E. et al. "Computer-based Consultations in Clinical Therapeutics - Explanation and Rule Acquisition Capabilities of the MYCIN System," Computers and Biomedical Research, vol. 8, p. 303-320, 1975.
2. Winston, P. and Horn, B. Lisp, Addison-Wesley Publishing Co., Reading MA, 1981.
3. McDermott, J. and Forgy, C. "Production System Conflict Resolution Strategies," in Pattern Directed Inference Systems, D. Waterman and F. Hayes-Roth (Eds.), Academic Press, New York, NY, p. 177-202, 1978.
4. Weiss, S. and Kulikowski, C. Designing Expert Systems, Rowman and Allanheld, Publishers, Totowa, New Jersey, 1984.
5. Chandrasekaran, B. and Mittal, S. "Deep Versus Compiled Knowledge Approaches to Diagnostic Problem-Solving," AAAI2, p. 249-354, 1982.
6. Cross, S. Qualitative Reasoning in an Expert System Framework, PhD Dissertation, Coordinated Science Laboratory T-124, University of Illinois, 1983.
7. Kunz, J. et al. "A Physiological Rule-based System for Interpreting Pulmonary Function Test Results," HPP No. 78-19, Computer Science Department, Stanford University, 1978.
8. Michie, D. Private conversation. 1981.
9. Davis, R. Use of Meta Level Knowledge in the Construction, Maintenance, and Use of Large Knowledge Bases, PhD Dissertation, Stanford University, STAN-CS-76-552, DTIC No. AD-76-25988, 1976.
10. Aikens, J. Prototypes and Production Rules: Knowledge and Representation for Consultation, PhD Dissertation, Stanford University, STAN-CS-80-814, 1980.
11. Doyle, J. "A Truth Maintenance System," Artificial Intelligence, vol. 12, p. 231-272, 1979.
12. Hayes-Roth, F., Waterman, D., and Lenat, D. Building Expert Systems, Addison-Wesley Publishing Co., Reading, MA, 1983.
13. Brooke, D. Natural Language Processing as a Universal Front-End to Expert Systems, MS Thesis, Air Force Institute of Technology, 1983.
14. Erman, L. and Lesser, V. "The Hearsay II Speech Recognition System," Lea (Ed.), Trends.
15. Forgy, C. The OPS5 Users Manual, Carnegie-Mellon University,

TR-CMU-CS-81-135, 1981.

16. McDermott, J. "An Expert in the Computer System Domain," AAAI, p. 269-271, 1980.

17. McArthur, D. and Klahr, P. "The ROSS Language Manual," N-1854-AF, The Rand Corp., Santa Monica, CA, 1982.

18. Fain, J. et al. "The ROSIE Language Reference Manual," Report No. N-1647, Rand Corp., Santa Monica CA, 1981.

19. Feigenbaum, E. and McCorduck, P. The Fifth Generation, Addison-Wesley Publishing Co., Reading MA, 1983.